REQ. NO. 0-9131

UNIVERSITY OF OTTAWA

DEPARTMENT OF MECHANICAL ENGINEERING

CONTRACT REPORT NO. OSU81-00137

MATHEMATICAL MODELLING OF HEAT TRANSFER IN PERMANENT MOLDS

(Rep. II - Cylindrical - 2D. Coordinates)

BY

DR. M. SALCUDEAN - RESEARCH COORDINATOR

A. MASHAIE - RESEARCH ASSOCIATE

July 1, 1983

124 pp

ABSTRACT

A mathematical model simulating the solidification of castings from pouring through freezing was constructed. The process is governed by a set of Partial Differential Equations. The Finite difference approximation was used to create a system of linear algebraic equations from the set of partial Differential equations. This was done using the energy balance method. An Alternating Direction Implicit (ADI) method was ued to solve the governing differential equations.

The tridiagonal system of algebraic equations was solved using the tridiagonal matrix algorithm (TDMA). A flow chart has been developed and a FORTRAN program was written for the numerical scheme. Preliminary testings of the model was done for a Simple Two-Dimensional solidification problem in cylindrical polar coordinates. The temperature distributions and profiles in the casting and mold, solidification front profiles and solidification time were computed. The model was compared with data from the literature.

TABLE OF CONTENTS

		Page
ABSTRACT		i
TABLE OF CONTENTS		ii
LIST OF FIGURES		iv
LIST OF TABLES		vi
NOMENCLATURE		vii
CHAPTER 1:	INTRODUCTION	1
1.1	The Solidification problem and its practical importance	1
1.2	Solidification of casting and ingots	2
1.2.1	Casting processes in which interface resistance is dominant	10
1.2.2	Analytic Solutions for ingot casting	12
1.3	Review of previous works	17
1.4	Objective and outline of present investigation	19
CHAPTER 2:	MATHEMATICAL FORMULATION	20
2.1	Introduction	20
2.2	General Equation of heat flow	20
2.3	Initial and Boundary Conditions	21
2.4	Thermal properties	24
2.5	LATENT Heat of Fusion	25
2.6	Discretization of the structure	29
2.7	Development of the Finite Difference Equation in Two-Dimensional cylindrical polar coordinates	29
2.7.1	Energy Balance Method	29
2.7.2	Finite Difference Equation for an Internal node	32
2.7.3	Finite Difference Equations for a Boundary node	32
2.7.4	Finite Difference Equation for a Center Line node	39
2.7.5	Finite Difference Equation for an Interface node	41
2.7.6	Identification of Finite Difference Equation for Typical Nodes	46

		Page
CHAPTER 3:	NUMERICAL PROCEDURES	48
3.1	Introduction	48
3.2	Outline of the Alternating Direction Implicit (ADI) Method	49
3.3	Computer program	51
3.4	General Remarks	54
CHAPTER 4:	PRELIMINARY TESTING OF THE MODEL	56
4.1	Introduction	56
4.2	Computational Details	56
	4.2.1 Problem statement	56
	4.2.2 Solution procedures	58
4.3	Sample Results and discussion	58
CHAPTER 5:	CONCLUSIONS	75
BIBLIOGRAPHY		76
APPENDIX		79

LIST OF FIGURES

FIGURE:		PAGE
1.1	Temperature profile in solidification of a pure metal |8|.	3
1.2	Approximate temperature profile in solidification of a pure metal poured at its melting point against a flat, smooth mold wall |8|.	4
1.3	Distance solidified versus square root of time for several pure metals in insulating molds |8|.	7
1.4	Chvorinov's experimental results on solidification time of castings versus their volume-to-area ratio |8|.	9
1.5	Temperature profile during solidification against a large flat mold wall with mold-metal interface resistance controlling |8|.	11
1.6	Temperature profile: Solidification against a flat mold wall when (a) resistance of solidifying metal is controlling and when (b) combined resistances of metal and mold are controlling |8|.	13
1.7	Unidirectional solidification of pure iron against three types of chills. No mold-metal interface resistance |8|.	15
1.8	Unidirectional solidification of pure iron against a cast-iron chill (a) No interface resistance, (b) carbon-black chill wash (c) thick insulating wash.	16
2.1	Correlations for the thermal conductivity of low carbon steel |17|.	26
2.2	Selected data and functional relationships for specific heat of low carbon steel, including the latent heat approximation |17|.	27
2.3	A three-dimensional element	30
2.4	Finite difference grid used in computations	31
2.5	Arrangement of grid points	33
2.6	Typical Internal node	34
2.7	Typical Boundary node	38
2.8	Typical Centerline node	40
2.9	Typical Centerline node (top corner)	42
2.10	Typical Centerline node (bottom mold corner)	43

		Page
2.11	Typical Interface node	45
2.12	Numbers indicating the type of node in the grid arrangement	47
3.1	Illustration of line-by-line iteration used in the ADI method	52
3.2	Flow chart for calculating solidification in a casting	53
4.1	Finite difference grid and boundary conditions	57
4.2	Temperature distribution at time = 30.0 minutes	61
4.3	Advance of Solidification front in the ingot	62
4.4	Solidified thickness versus square root of time	63
4.5	Isotherms in the ingot at (a) time = 10.0 min. and (b) time = 40.0 min. after pouring	64
4.6	Isotherms in the mold at (a) time = 10.0 min. and (b) time = 40.0 min. after pouring	65
4.7	Temperature gradients in the ingot and mold at time = 10.0 min. after pouring	66
4.8	Temperature gradients in the ingot and mold at time = 40.0 min. after pouring	67
4.9	Cooling curves at the centre of ingot	69
4.10	Ingot and mold temperature - time variation curves at 66.7 cm from the top	70
4.11	Variation of heat transfer coefficient at the side outer surface of mold with time	72
4.12	Variation of heat transfer coefficient at the bottom outer surface of mold with time	73
4.13	Variation of the interface heat transfer coefficient	74

LIST OF TABLES

TABLE:		PAGE
4.1	Input data used in testing the program	59

NOMENCLATURE

a_n, b_n, c_n, d_n	-	Coefficients in a tridiagonal system of equating (3.4)
C, C_p	-	specific heat
CYL	-	Binary variable taking zero or unity
DT	-	equivalent temperature defined by equation (2-13)
erf	-	error function defined in equations (1-2) & (1-6) through (1-19)
f	-	interpolation factor (equation 2-16)
F	-	geometric factor
h	-	heat transfer coefficient
h_s, h_b, h_g	-	heat transfer coefficients at side and bottom boundary, and air-gap respectively
h^*	-	equivalence heat transfer coefficient accounting for the conduction heat flow at the contact spots
H	-	latent heat of fusion
K	-	thermal conductivity
$\frac{\partial T}{\partial n}$	-	partial derivative of temperature in the normal direction
Q	-	internal energy generated per unit volume per unit time
r, θ, z	-	cylindrical coordinates
$\frac{\partial T}{\partial r}, \frac{\partial T}{\partial o}, \frac{\partial T}{\partial z}$	-	partial derivatives of temperature in r, θ and z directions respectively
t	-	time
T	-	temperature
T^*	-	intermediate temperature at the end of first $\Delta t/2$ (ADI) method)
T_s	-	solidus temperature
T_L	-	liquidus temperature
T_{mold}, Tmelt		initial mold and metal temperature respectively
x, y, z	-	cartesian coordinates
$\frac{\partial T}{\partial x}, \frac{\partial T}{\partial y}, \frac{\partial T}{\partial z}$	-	partial derivatives of temperature in x, y, z direction

GREEK SYMBOLS:

α	thermal diffusivity $(= \frac{K}{\rho C_p})$
Δ	increment
ε	emissivity
ρ	density
σ	Stefan-Boltzmann constant

SUBSCRIPTS:

a	ambient
b	boundary
eff	effective
i,k	grid point subscript in x and z direction respectively
int	interface
s	mold
m	metal

SUPERSCRIPTS:

$\eta+1$	quantities evaluated at $(\eta+1)$th time step

CHAPTER 1

INTRODUCTION

1.1 THE SOLIDIFICATION PROBLEM AND ITS PRACTICAL IMPORTANCE

Knowledge of solidification rate and of the temperature contours in the ingots and castings are important to the process metallurgists. For example, if one knows how fast a cast ingot will solidify right through, optimization of the stripping time, soaking pit times, and soaking pit temperatures can be carried out. Segregation, inclusions, porosity, etc. are matters of general interest since they exert an adverse influence on material properties.

In foundry practice, after the furnace operations have been completed, the molten metal leaving the furnace is tapped into a container called a ladle and from the ladle, is poured into molds that have the size and shape of the castings desired. The solidification process that takes place in the molds involves many complex phenomena: temperatures change with heat flow as a function of time, internal heat is generated during the phase change and varying conditions are found at the boundaries.

The rate of solidification depends on the size of the casting, quality of the materials involved, pouring temperature, rate of pouring and the design of the mold, therefore, the control of the whole solidification process is very important.

The process of solidification can be described in terms of partial differential equations. These equations, with the appropriate boundary conditions representing the actual foundry operations, can be solved using numerical techniques. While numerical calculation schemes probably never substitute entirely for the experimental investigations. They have already been proven useful for exploring and helping to optimize fairly complicated heat transfer systems in which measurements are difficult, costly, and laborious to obtain |28|.

1.2 SOLIDIFICATION OF CASTING AND INGOTS

GENERAL THEORY:

In most casting and ingot-making processes, heat flow is not at steady state. Hot liquid is poured into a relatively cold mold; specific heat and heat of fusion of the solidifying metal pass through a series of thermal resistances to the cold mold until solidification is complete. Figure (1-1) shows this process schematically for solidification of a pure metal.

Thermal resistances which, in general, must be considered are those across the liquid, the solidifying metal, and the metal-mold interface and those in the mold itself. The problem is mathematically and physically complex and becomes even more so when anything other than simple geometries are considered. Further, the complexity of the analysis of the processes increases when the mold and ingot are comprised of multi-materials, and also when thermal properties are allowed to vary with temperature, or when alloys are considered [8]. There are, however, certain simplifying approximations that can be made for a number of cases of engineering interest. Sand casting is a process for making shaped castings which employ relatively insulating molds.

From a heat-flow stand point, the important characteristic of solification of a metal in a sand mold is that the metal is a much better conductor of heat than a sand mold. Thus, solidification rate is dependant on thermal properties of the mold. In many cases, the mold can be considered to be semi-infinite in extent; i.e., the outside of the mold does not heat-up during solidification. The heat flow problem sketched in Fig. (1-1) is now very much simplified, especially if we assume further that the metal is poured with no super-heat, that is, exactly at its melting point T_{melt}, as shown in Fig. (1-2).

Without loss of generality, we consider the problem of unidirectional heat flow. Metal is poured exactly at its melting point against a thick, flat mold wall initially at the temperature T_{mOld} .

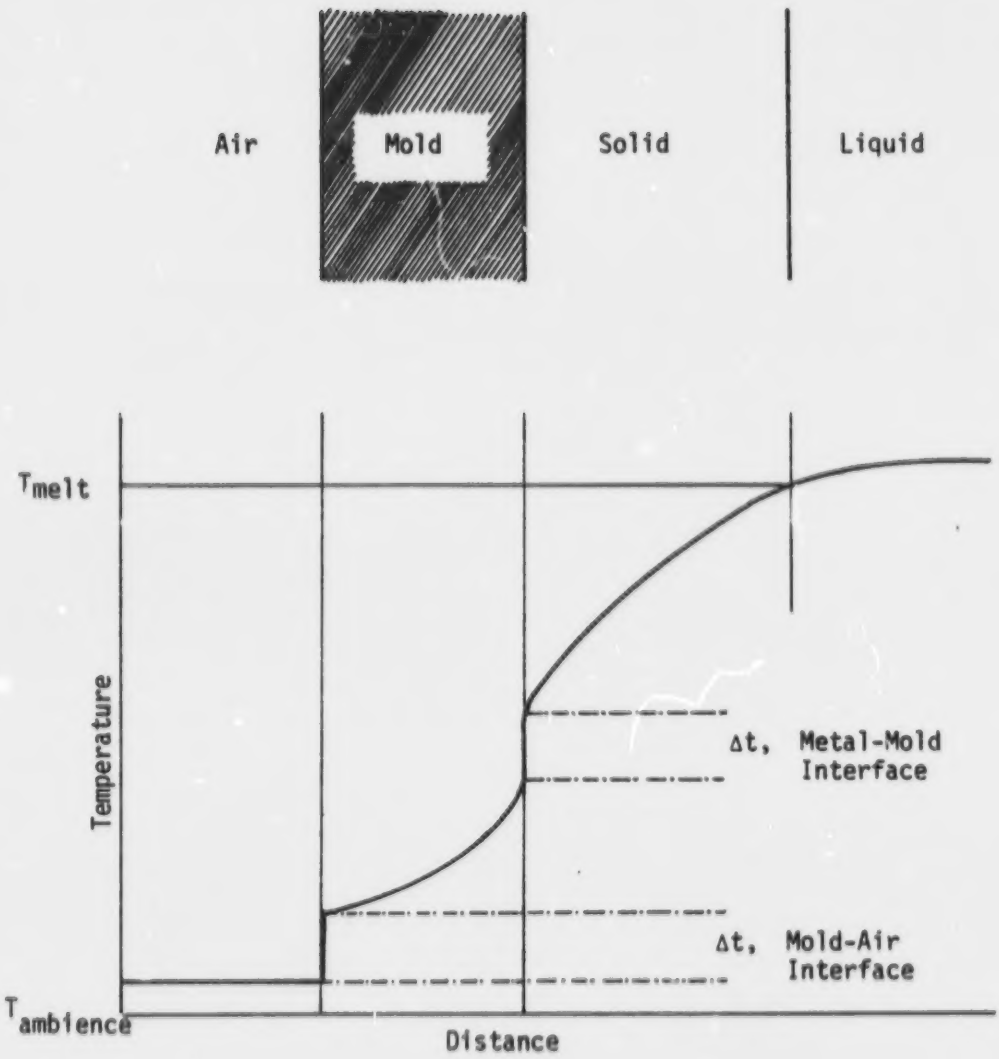


Figure 1-1
Temperature profile in solidification of pure metal |8|.

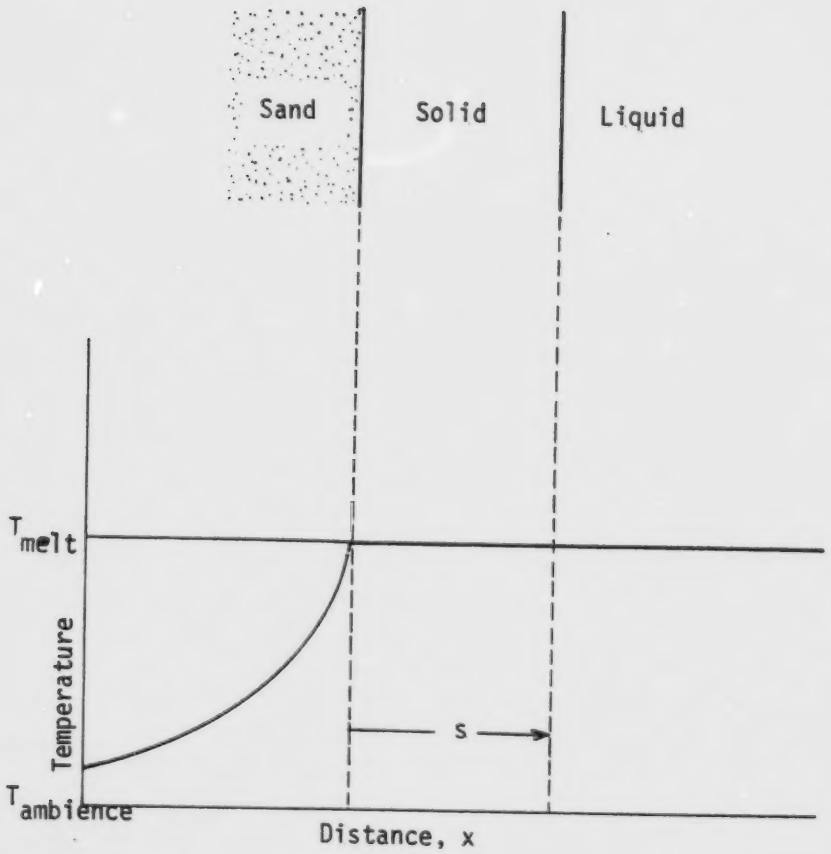


Figure 1-2
Approximate temperature profile in solidification of a pure metal poured at its melting point against a flat, smooth mold wall.

Thus, the mold surface is heated suddenly to T_{melt} at time $t = 0$. (This assumption is not applicable in our case, see section 2.3). This is a transient, one-dimensional heat-flow problem, and the solution must conform with the partial differential equation

$$\frac{\partial T}{\partial t} = \alpha_m \frac{\partial^2 T}{\partial x^2} \qquad (1\text{-}1)$$

where α_m = Thermal diffusivity of mold, cm^2/s

K_m = Thermal conductivity of mold, $Cal/(cm)(^oc)(s)$

ρ_m = Density of mold, g/cm^3

t = time, second

x = Distance from mold wall, cm (negative into the mold)

The solution to this equation for the boundary conditions stated above gives the temperature T in the mold as a function of time t at distance from the mold surface x:

$$\frac{T - T_{melt}}{T_{mold} - Tmelt} = \text{erf} \frac{-x}{2\sqrt{\alpha_m t}} \qquad (1\text{-}2)$$

where "erf" denotes the error function. The error function of zero is zero, and the error function of infinity is unity.
The rate of heat flow into the mold at the mold-metal interface is given by

$$\left(\frac{q}{A}\right)_{x=o} = -K_m \left(\frac{\partial T}{\partial x}\right)_{x=o} \qquad (1\text{-}3)$$

where x increases positively from left to right in Figure (1-2), q is the rate of heat flow, and A is the area of the mold-metal interface. By partial differentiation of Equation (1-2), with respect to x, letting $x = o$ and combining the results with Equation (1-3), the rate of heat flow across the mold-metal interface is seen to be:

$$\left(\frac{q}{A}\right)_{x=o} = -\sqrt{\frac{K_m \rho_m C_m}{\pi t}} (T_{melt} - T_{mold}) \qquad (1\text{-}4)$$

where C_m is specific heat of the mold material. Now, the heat entering the mold comes only from the heat of fusion of the solidifying metal since the solid as well as the liquid metal is exactly at T_{melt} (Fig. 1-2).

Thus,

$$\left(\frac{q}{A}\right)_{x=o} = -\rho_s H \frac{\partial s}{\partial t} \qquad (1\text{-}5)$$

where s = thickness solidified. Combining (1-4) and (1-5) and integrating from s = o at t = o,

$$S = \frac{2}{\sqrt{\pi}} \underbrace{\left(\frac{T_{melt} - T_{mold}}{\rho_s H}\right)}_{\text{Metal}} \underbrace{\sqrt{K\rho mCm}}_{\text{Mold}} \sqrt{t} \qquad (1\text{-}6)$$

Equation (1-6) predicts the manner in which thermal properties of the metal and the mold combine to determine the freezing rate of a metal casting into a relatively insulating mold. The relationship is more accurate for sand castings of high conductivity metals, such as nonferrous metals (cu-, Aℓ-, and Mg- base alloy), than for iron and steel. Note that high melting temperature and low heat of fusion favor rapid solidification, |8|. The product KmρmCm is a measure of the rate at which the mold can absorb heat.

The thickness of solid metal is a parabolic function of time, which means the solidification rate is initially very rapid and decreases as the mold becomes heated. Figure (1-3) shows the wide range of solidification rates attained in practice, depending on metal and mold temperatures |8|.

The one-dimensional freezing problem serves to illustrate many of the important thermal aspects of solidification, but for some purposes it is important to evaluate freezing times and rates of complex shapes. Consider again the question of heat flow into a mold wall. The contour of the mold wall has some influence on its ability to absorb heat. For example, the geometry of heat flow into a concave or convex contoured mold wall may be compared with that into a plane mold wall. Heat flow into the concave surface will be divergent and therefore slightly more rapid, and into the convex surface less rapid than into a plane wall. For simple shapes, however, the differences will not be large and a useful approximation is to assume that a given square centimeter of mold surface has a fixed ability to absorb heat regardless of its contour or location on the casting. With this approximation, we may now replace S in equation (1-6) with $V_{s/A}$ where Vs is volume solidified at time t and A is area of the mold-metal interface. Or, letting $t = t_f$, where t_f is the total solidification time of a casting of volume V,

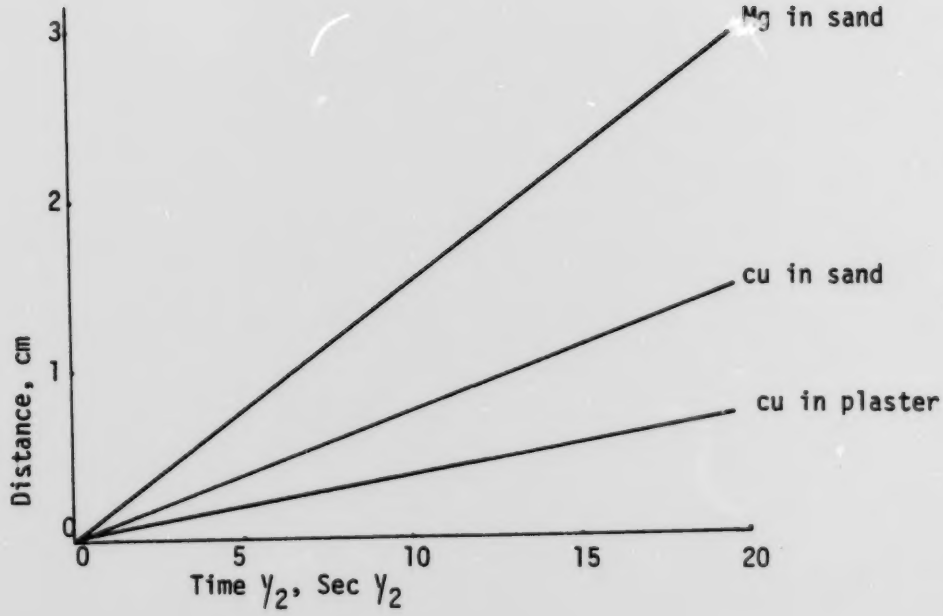


Figure 1-3

Distance solidified versus square root of time for several pure metals in insulating molds.

$$\frac{V}{A} = \frac{2}{\sqrt{\pi}} \left(\frac{T_{melt} - T_{mold}}{\rho_s H}\right) \sqrt{K_m \rho_m C_m} \sqrt{t_f} \qquad (1\text{-}7)$$

and $$t_f = c\left(\frac{V}{A}\right)^2 \qquad (1\text{-}8)$$

where c is a constant for a given metal-mold material and mold temperature.

Equation (1-8) is the well known Chvorinov's rule used to compare solidification times of simple-shaped castings. It states that the total time of solidification of such castings is proportional to the square of the volume-to-area ratio of the castings. Experimental confirmation of this result is seen in original experiments of Chvorinov on steel castings of different sizes and shapes, varying from 10mm thick to a 65-ton casting (Figure 1-4). Chvorinov's times of complete solidification were obtained from thermo-couple measurement at the thermal center of the casting |8|. For shapes such as spheres or cylinders, it is possible to derive a more exact expression that Equation (1_7) relating t_f to V/A without retaining the assumption of non-divergency of heat flow. In these cases, the applicable partial differential equation for the heat flow in the mold is

$$\frac{\partial T}{\partial t} = \alpha_m \left(\frac{\partial^2 T}{\partial r^2} + \frac{n}{r} \frac{\partial T}{\partial r}\right) \qquad (1\text{-}9)$$

where r is the casting radius and n = 1 for cylinder, 2 for sphere. Following a procedure identical to that used to derive Equation (1-7), the resulting equivalent expression is

$$\frac{V}{A} = \frac{T_{melt} - T_{mold}}{\rho_s H} \left(\frac{2}{\sqrt{\pi}} \sqrt{K_m \rho_m C_m} \cdot \sqrt{t_f} + \frac{nK_m t_f}{2r}\right) \qquad (1\text{-}10)$$

By comparison of (1-10) with (1-7) it may be seen that the simple Chvorinov approximation becomes increasingly valid as thermal diffusivity $K_m/\rho_m C_m$ decreases. It is also more nearly valid for cylinders than for spheres. For a given volume-to-surface area ratio, a sphere freezes more rapidly than a cylinder and a cylinder more rapidly than a plate.

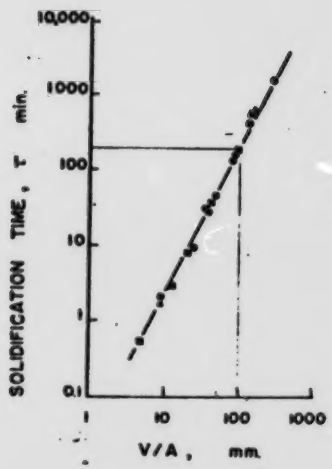


Figure 1-4

Chvorinov's experimental results on solidification time of castings versus Their Volume-to-area rate |8|.

1.2.1 CASTING PROCESSES IN WHICH INTERFACE RESISTANCE IS DOMINANT

In a large number of important casting processes, heat flow is controlled to a significant extent by resistance at the mold-metal interface. These processes include the permanent-mold-casting, die casting, splat-cooling, |8|, and powder manufacturing processes. When the mold-metal interface resistance is of overriding importance, the temperature distribution across the solidifying metal and mold is as in Figure (1-5). All temperature drop is across the interface. The mold, being assumed infinite in extent, remains at its original temperature T_{mold}.

The rate of heat flow across this interface for metal poured at its melting point T_{melt} is

$$\left(\frac{q}{A}\right)_{x=o} = -h(T_{melt} - T_{mold}) \qquad (1\text{-}11)$$

combining (1-11) with (1-5) for the case of a large flat mold wall and integrating from $s = o$ at $t = o$. Yields

$$s = h \frac{T_{melt} - T_{mold}}{\rho_s H} t \qquad (1\text{-}12)$$

Furthermore, since shape in no way alters the heat transfer across the interface, Equation (1-12) may be generalized for simple-shaped castings to calculate the solidification time t_f in terms of the volume-to-area ratio of the casting:

$$t_f = \frac{\rho_s H}{h(T_{melt} - T_{mold})} \cdot \frac{V}{A} \qquad (1\text{-}13)$$

Equations (1-12) and (1-13) are valid when the resistance to heat flow across the mold-metal interface is large compared with other resistances in the metal and mold.

Except when the mold is relatively insulating, this condition pertains when

$$h \ll \frac{Ks}{S}$$

when the mold is relatively insulating, there is the added necessary condition that

$$h^2 \ll \frac{K_m \rho_m C_m}{t} .$$

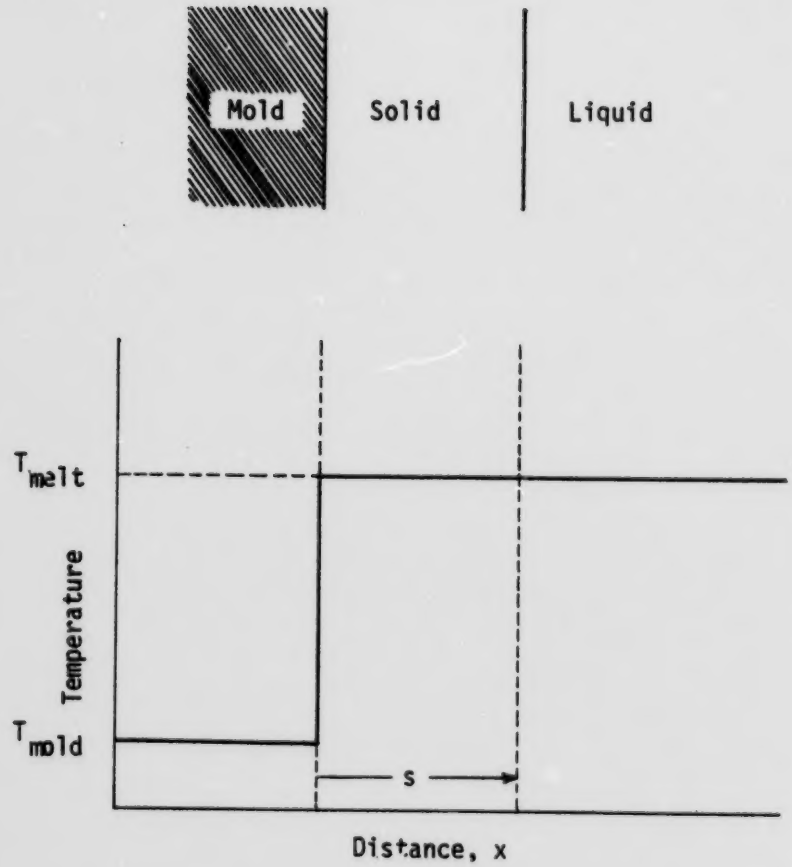


Figure 1-5

Temperature profile during solidification against a large flat mold wall with mold metal interface resistance controlling.

1.2.2 ANALYTIC SOLUTIONS FOR INGOT CASTING

For many important ingot-casting processes, the thermal profile sketched in Figure (1-1) applies and the thermal resistances of the metal, mold-metal interface, mold, and mold surroundings must be considered for complete solution. Exact analytic solutions are available for limited cases of engineering interest where heat flow is one-dimensional, mold-metal interface resistance is negligible, and the mold either is held at constant temperature (as by water cooling) or is very thick. Temperature distributions for these cases are sketched in Figure (1-6).

Consider, first, the case of the water-cooled chill. The solution to this problem must conform with the applicable partial differential equation [Equation (1-1)], with the boundary conditions that at $x = o$, $T = T_{mold}$, at $x = s$, $T = T_{melt}$, and at the liquid-solid interface

$$K_s\left(\frac{\partial T}{\partial x}\right)_{x=s} = H\rho_s \frac{\partial S}{\partial T} \qquad (1\text{-}14)$$

The solution, given by CarsLaw, and Jaeger is

$$S = 2\gamma\sqrt{\alpha_s t} \qquad (1\text{-}15)$$

γ is determined from

$$\gamma e^{\gamma^2} \operatorname{erf}\gamma = (T_{melt} - T_{mold})\frac{C_s}{H\sqrt{\pi}} \qquad (1\text{-}16)$$

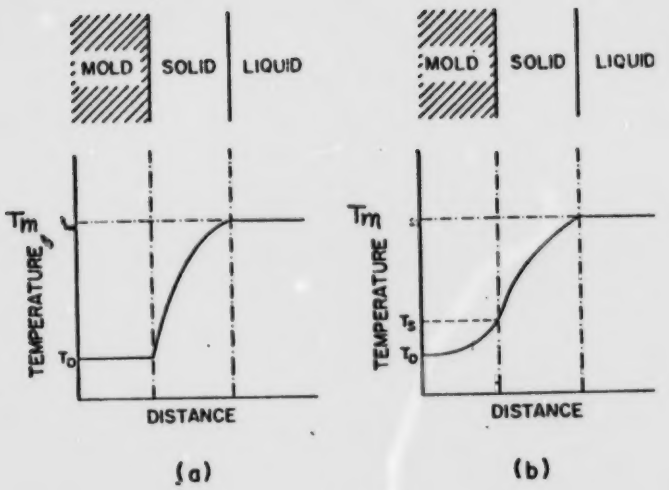
The temperature distribution in the solidifying metal is given by

$$\frac{T - T_{mold}}{T_o' - T_{mold}} = \operatorname{erf}\frac{x}{2\sqrt{\alpha_s t}} \qquad (1\text{-}17)$$

where T_o' is the constant of integration.

The solution when the mold is not water-cooled (but is Semi-infinite) is only a little more complicated. It is

$$\gamma e^{\gamma^2}\left(\sqrt{\frac{K_s\rho_s C_s}{K_m\rho_m C_m}} + \operatorname{erf}\gamma\right) = \frac{Cs}{H\sqrt{\pi}}(T_{melt} - T_{mold}) \qquad (1\text{-}18)$$



$T_o = T_{mold}$

$T_m = T_{melt}$

$T_s = T_{int}$ (Temperature at interface)

Figure 1-6

Temperature profile: Solidification against a flat mold wall when (a) resistance of solidifying metal is controlling and when (b) combined resistances of metal and mold are controlling |8|.

Equation (1-17) must now be rewritten as, replacing T_{mold} by the temperature of the mold-metal interface T_s,

$$\frac{T - T_s}{T_o - T_s} = \text{erf} \frac{x}{2\sqrt{a_s t}} \qquad (1\text{-}19)$$

and Equation (1-15) applies for this case as before. Thus, Equation (1-15), (1-18), and (1-19) constitute the solution to the heat flow problem for unidirectional solidification in a non-water-cooled metal mold. These equations, as well as equation (1-17) for the water-cooled chill and equation (1-6) for the insulating mold, represent special cases of Schwartz' solution (for zero superheat). Note that for both the water-cooled and non-water-cooled chill, solidification is parabolic with time. This result has been observed experimentally by a large number of investigators studying rate of solidification in steel ingots. Some typical calculated results are shown in Figure (1-7). These results agree well with experimental measurements made of unidirectional solidification against metal chill walls except that the experimental curves obtained are generally displaced slightly on the time axis |8|.
As an example, many investigators have obtained experimental results for steel solidification against cast-iron-mold walls which are of the form

$$S = a\sqrt{t} - b \qquad (1\text{-}20)$$

where the constants a and b are approximately unity and 0.5, respectively, when the units are inches and minutes (as in Figure (1-7). The apparent delay in the beginning of the solidification found experimentally is probably caused by: a) convection during and just after pouring resulting in rapid removal of superheat from the liquid, thus slowing the start of solidification and its initial rate, b) the second effect is finite mold-metal resistance to heat transfer. In this case the solidification rate is actually finite at zero time but is interface-controlled and so proceeds at a slower rate than in the absence of this resistance. Initially, the amount solidified increases linearly with time, as given by Equation (1-12). Later, as the resistance of the solidifying metal becomes large compared with mold-metal resistance, the amount solidified increases linearly with the square root of time as shown in Figure (1-8).

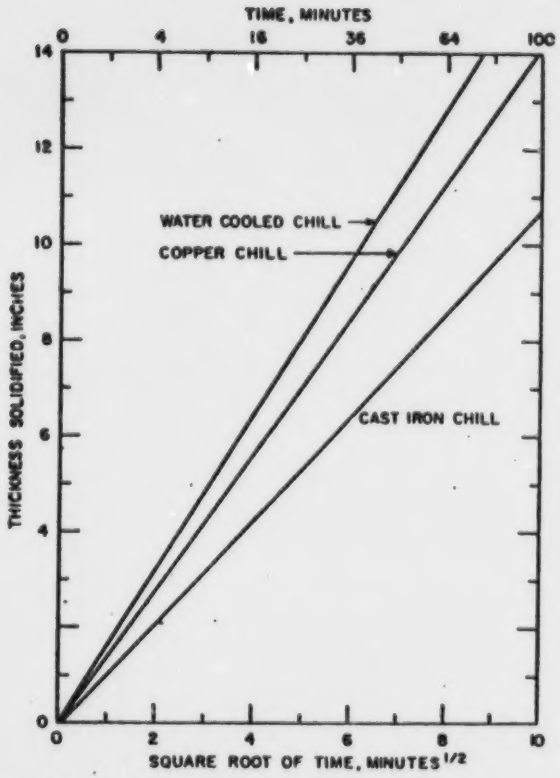


Figure 1-7

Unidirectional solidification of pure iron against three types of chills. No mold-metal interface resistance |8|.

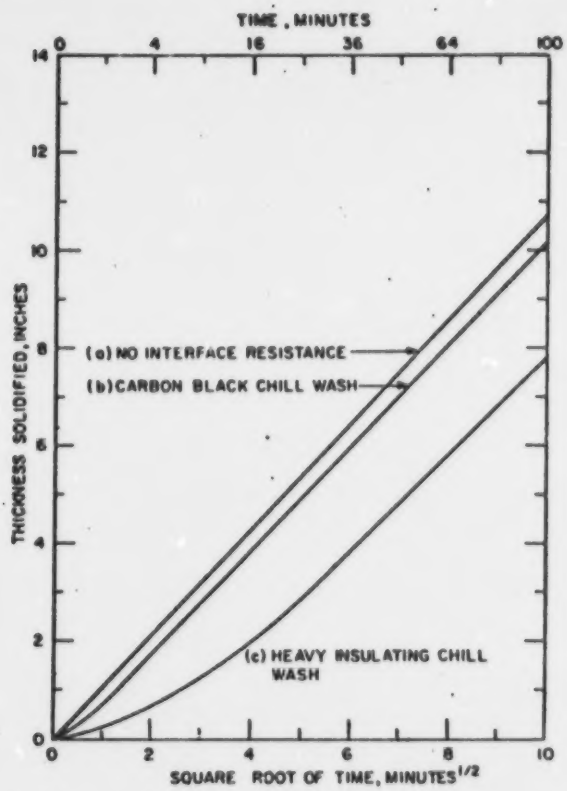


Figure 1-8

Unidirectional solidification of pure iron against a cast-iron chill.
(a) No interface resistance
(b) Carbon-black chill wash
(c) Thick insulating wash |8|.

Approximate analytical solutions for treating this problem have been given by several investigators and compared by Jones. Figure (1-8) shows results for pure iron with no superheat using Adam's method and values of the mold-metal heat-transfer coefficient that are in the range of those encountered in practice. Note that in the case of iron the interface resistances have relatively little effect on solidification behaviour except very near the chill. Similar interface resistances have a relatively large effect with high conductivity metals such as aluminum since the distance from the chill at which they will remain of importance depends on the ratio of the interface resistance to the metal thermal conductivity.

In all the solutions given in the foregoing paragraphs, metal superheat has been neglected. Exact analytical solutions are available to account for this superheat in simple one-dimensional solidification (in absence of interface resistance) providing thermal properties are independent of temperature and thermal convection is absent. However, it is clear that even very small horizontal temperature gradients result in vigorous convection in liquid metals and, therefore, that the superheat is dissipated early in solidification. Experimentally also it is well established that the superheat is nearly or completely eliminated in usual ingot - and shaped-casting processes very soon after pouring. In sand castings or in the case of interface - controlled solidification the effect of superheat on the time to complete solidification can be approximately accounted for by adding the heat content due to superheat to that of the heat of fusion.

1.3 REVIEW OF PREVIOUS WORKS

Studies of the solidification process date back to the 1920's |7,4|. The last few decades have seen many theoretical analyses using mathematical models to complement experimental investigations of the solidification phenomena.

Analytical solutions have been developed for various heat flow problems. Analytical solutions, however, are highly restrictive and often simplified. As such, the results obtained are only indicative. Some of the earlier works were by Lightfoot |15,16| and Adams |1|. More recent studies were done by Garcia and Prates |9,10| to predict the solidification rate and the temperature distribution during the unidirectional solidification of metals in cooled and uncooled molds.

Paschkis |21-23| simulated the solidification process by using analog computers. His early works contributed much to many improvements in the design concepts for castings. The analog computer, however, is limited to simple problems. Numerical techniques using the digital computer permits simulation of a more realistic problem without simplification.

In earlier works, Sarjant and Slack |29| and Dusinberre |6| used a technique developed for the numerical integration of the heat-flow equations in two dimensions. The main advantage of the Sarjant and Slack method is that, by using an enthalpy term, the latent heat of solidification is taken into account in the calculations. However, the method requires very small time-steps to satisfy the stability criterion. Nishida et al |20|, Srinivasion |32| and Lazaridis et al |14| used the explicit finite difference scheme for predicting one-dimensional solidification problem with constant thermal properties. The Crank-Nicolson finite difference method was used by Thamban and Panchanathan |34| for simulating unidirectional solidification of aluminium alloys in cast iron molds. The thermal properties of pure aluminium castings was assumed to vary with temperature and the formation of an air-gap was considered.

Marrone, Wilkes and Pehlke |17,18| investigated the solidification of low-carbon steel casting in simple "T" and "L" shaped sand molds. The Saul'yev explicit finite difference algorithm and the implicit alternating direction were used to solve the heat flow equation. In addition, a technique has been developed for determining material thermal properties. This work was continued by Pehlke et al |26| which involves the comparison of a computer simulation utilizing independently determined thermal properties with result of a thermally monitored casting. Raychandhuri et al |27| simulated two-dimensional solidification by describing the solidification process in three stages.

Additional equations and boundary conditions were derived to satisfy the conditions at each stage. A number of other studies of solidification using finite difference approximation have also been published |2,3,5,13,25,31,33|.

While finite difference methods have been the most popular methods for numerical solution of solidification problems, a number of models have been developed using the finite element method |11,12,19,35,36|.

A list of references does not exhaust the topics of mathematical modelling of heat flow and solidification. Unfortunately, most of the published works deal with one- or two-dimensional problems and their lack of reliable information on thermal properties requires the use of many simplifying assumptions.

1.4 OBJECTIVE AND OUTLINE OF PRESENT INVESTIGATION

The main objective of this investigation is to develop a mathematical model aimed at increasing the understanding of the solidification process and obtaining relevant operating and materials parameters which can be useful for the process design and control. This includes the determination of solidification time, solidification front profiles, and temperature distribution in castings and molds for the further studying of the thermal stress in the molds. In essence, this work involves the mathematical modelling of the solidification of castings in cylindrical molds. Applying the finite difference technique, the set of partial differential equations governing the processes generate a system of linear algebraic equations. The alternating direction implicit technique is applied to solve the above system of equations.

The cases considered include metal castings in metal molds. Future developments will include different metals and alloy castings in water-cooled molds, composite molds, sand molds or molds with complex shapes which might require three-dimensional analysis. The release of latent heat of fusion and gap formation is also modelled. The variation of the thermal properties with temperature is handled by extrapolation or correlations obtained from available data which can be easily implemented in the finite difference approximations.

CHAPTER 2

MATHEMATICAL FORMULATION

2.1 INTRODUCTION

The solidification process is basically a problem of unsteady-state heat flow in three dimensions. The modelling of the solidification process, however, is one of considerable mathematical complexity. As solidification takes place, heat transfer occurs in three ways - conduction, convection, and radiation. These are defined by the Fourier, Newton, and Stefan-Boltzmann laws, respectively. Numerical solutions of the differential equations derived from these laws can be obtained by finite-difference approximations. In most casting and ingot-making processes, the solidification of the metal involves a phase change which results in the release of latent heat of fusion. The problem becomes even more complex when the thermal properties of the mold and metal are temperature-dependent, or when solidification of alloys or metals with crystals of complex geometries are considered. Some of these problems are considered and discussed in this chapter. There are, however, certain simplifying approximations which are made to handle a number of cases which are of interest.

2.2 GENERAL EQUATION OF HEAT FLOW

The unsteady-state heat flow equation is derived by considering the energy balance on a differential element of volume of the material subject to a conduction process. The sum of the total heat absorbed by the element per unit time due to heat flow and the internal heat generated in the element per unit time is equal to the net increase of internal energy of the element per unit time. The general equation of heat conduction, written in three-dimensional cartesian coordinates is,

$$\frac{\partial}{\partial x}\left(K\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(K\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(K\frac{\partial T}{\partial z}\right)+Q=\rho C_p\frac{\partial T}{\partial t} \qquad (2\text{-}1)$$

The first three terms on the LHS represent the net energy gain by conduction per unit time across the element surfaces.

Q is the energy generated in the element per unit time and the term on the RHS is the change of energy stored in the element per unit time. Often, practical problems in solidification processes involve cylindrical geometries. Hence the general equation written in cylindrical coordinates is

$$\frac{1}{r}\frac{\partial}{\partial r}\left[rK\frac{\partial T}{\partial r}\right]+\frac{1}{r^2}\frac{\partial}{\partial \theta}\left[K\frac{\partial T}{\partial \theta}\right]+\frac{\partial}{\partial z}\left[K\frac{\partial T}{\partial z}\right] + Q = \rho C_p \frac{\partial T}{\partial t} \qquad (2\text{-}2)$$

Both equations (2-1) and (2-2) are written for variable conductivity, K which is discussed in Section 2.4.

2.3 INITIAL AND BOUNDARY CONDITIONS

The prediction of the temperature field in a solidifying casting is essentially that of solving the unsteady-state heat flow equation. In order to solve this equation, the boundary conditions which are characteristic of the process must be considered. Since the problem is one of unsteady-state heat transfer, an initial condition must also be specified.

At the surface boundary, two cases of boundary conditions are considered:

(a) Dirichlet known boundary condition in which case the boundary surface temperature is known or given in the problem specification

(b) Neumann derivative boundary condition in which case the boundary surface temperature is not known and the boundary specification is given in the form of information about the outward normal gradient of temperature $\frac{\partial T}{\partial n}$. This may be convective, and the outward normal temperature gradient is written as

$$\frac{\partial T}{\partial n} = -\frac{h}{k}(T_b - T_a) \qquad (2\text{-}3)$$

where h is the surface heat transfer coefficient. T_b is the boundary and T_a the ambient temperatures. The normal temperature gradient may take the form

$$\frac{\partial T}{\partial n} = -\frac{F\varepsilon\sigma}{k}(T_b^4 - T_a^4) \qquad (2\text{-}4)$$

for a radiative boundary, where F is a geometric factor, ε is the emissivity, and σ is the Stefan-Boltzmann constant.

If a particular boundary is known to be adiabatic, the outward normal temperature gradient is set to zero, so that

$$\frac{\partial T}{\partial n} = 0 \qquad (2\text{-}5)$$

Where heat transfer at the boundary is by both radiation and convection, the normal temperature is expressed as

$$\frac{\partial T}{\partial n} = -\frac{h}{k}(T_b - T_a) - \frac{F\varepsilon\sigma}{k}(T_b^4 - T_a^4) \qquad (2\text{-}6)$$

Equation (2-6) may be written as

$$\frac{\partial T}{\partial n} = -\frac{h_{eff}}{k}(T_b - T_a) \qquad (2\text{-}7)$$

where h_{eff} is the effective heat transfer coefficient of the surface which is calculated as follows

$$h_{eff} = h + F\varepsilon\sigma (T_b^2 + T_a^2)(T_b + T_a) \qquad (2\text{-}8)$$

Initially, the mold is assumed to be at a uniform temperature T_{mold}, and is assumed to be filled instantaneously with liquid metal at the pouring temperature T_{melt}. This assumption is valid only if the filling time is a small fraction of the total solidification time of the casting in which case, this assumption would not introduce any significant errors. For processes, however, where rapid solidification occurs or where a large temperature variation occurs during the filling, this assumption would not be valid. The actual filling procedure could be included in the model, but this would add a sophistication to the solution which is not necessary for the purpose of the investigation at the present stage. The pouring time is neglected and computations begin at time = 0. Once the mold is filled, it is further assumed that the liquid metal remains stagnant, with no convective transport in the liquid phase. Because of the complexity of the interaction between the thermally induced natural convective flow field and the moving solidification front, the convective transport is not included in the present studies. For future development of the model, the transient effects of solidification accompanied by convective transport may be included for more detailed studies.

Initially, heat is transferred from the liquid metal across the metal/mold interface, through the mold and to the surroundings. Neglecting any thermal contact resistance at the interface, the initial metal/mold interfacial temperature can be shown to be |18|,

$$T_{int} = T_{mold} + \frac{T_{melt} - T_{mold}}{1 + \frac{K_s}{K_m}\sqrt{\frac{\alpha_m}{\alpha_s}}} \qquad (2\text{-}9)$$

where α_m is the thermal diffusivity of the metal and α_s that of the mold.

At the metal/mold interface, the metal and the mold are initially in good thermal contact and heat transfer is by conduction. As the metal solidifies, it contracts and breaks away from the mold. Eventually an air-gap is formed between the solidified shell and the mold. The time of formation of the air-gap is one of the important controlling factors determining the quality of the casting. For the purpose of modelling, an air-gap formation time is assumed, depending on the cast size, shape and composition, type of mold and initial temperature of the system. The gap is assumed to form all over the interface at the same instant.

The air-gap constitutes a major resistance to the heat flow. The temperature distribution across the metal/mold interface indicates an apparent temperature discontinuity. When there is perfect contact, the continuity of heat flow should be maintained, therefore,

$$-K_m \frac{\partial T_m}{\partial n} = -K_s \frac{\partial T_s}{\partial n} \qquad (2\text{-}10)$$

But usually a discontinuity exists in the temperature gradient, since $k_m \neq k_s$ in general.

Across the air-gap, heat transfer is mainly by radiation. Due allowance may be made for convection and there may also be some spot contacts between the casting and the mold. Some form of artificial heat transfer coefficient may be necessary to account for these additional two modes of heat transfer mechanism, though in most cases, these may be neglected. However, there is no practical physical model of gap formation and heat transfer characteristics across narrow gaps at operating metallurgical temperatures.

Hence, an effective heat transfer coefficient for the air-gap was calculated based on the temperatures of the casting and mold across the air-gap. This is expressed as

$$h_g = h^* + F\varepsilon\sigma \ (T_m^2 + T_s^2)(T_m + T_s) \qquad (2\text{-}11)$$

where T_m and T_s are the temperatures of the casting and mold across the air-gap. h* is the equivalent heat transfer coefficient accounting for the conduction heat flow at the contact spots.

2.4 THERMAL PROPERTIES

In modelling the solidification process, thermal properties play an important role in achieving good simulation results. The simulated results are only as good as the data input. There are currently many publications and referenced sources with available thermal data. However, even with these sources, still no suitable data exists for the thermal properties especially for metals near their melting points or for liquid metals. Data at these high temperatures are scarce and often conflicting. This constraint is a limiting factor in the accuracy of the simulation.

Early analytical works [1,9,10,15,16] assumed constant thermal properties to obtain their solutions. Thermal properties like thermal conductivity, heat capacity, and density all vary with temperature to some degree. For a typical casting process, the temperature may change over a wide range. Thus, the analytical solutions obtained by oversimplifying this assumption are not realistic and hence do not have much practical use. To develop a model which promises to be useful, the thermal properties of the materials should be allowed to vary with temperature as they do in the real situation.

Thermal conductivities are very temperature-dependent. Sometimes they vary widely over the temperature range typically found in the casting process. Although values of thermal conductivities for typical engineering materials are available in the literature, the variations of thermal conductivities with temperature are not so readily available. Experimental determinations of thermal conductivity have been carried out for many engineering materials, most of them are for pure materials. For alloys, little has been done.

Until such data become available, it may be necessary to represent the variation of thermal conductivity by some correlations which follow the general trend as the available data for the materials involved. An example is illustrated in Figure (2-1). The thermal conductivity of low-carbon steel is represented by four linear segments over the temperature range. Faced with a wide spectrum of available thermal conductivity data, a polynomial regression was performed to choose the best quadratic curve fit based on the least square method. These curves appear to give reasonable results. The variation of density with temperature within a given phase (Liquid or Solid) is very small and is neglected in the present investigations, therefore, the density of materials involved within any phase is assumed to be constant. A representation of the specific heat is more complicated, because of the release of the latent heat of fusion when the molten metal solidifies. It will be dealt with in the next section. An illustration of the functional relationships for specific heat of low-carbon steel is shown in Figure (2-2).

The latent heat of fusion is accounted for, over the temperature range from the liquidus to the solidus. The area under the triangle is numerically equal to the latent heat of fusion.

Thus, any heat transfer analysis requires a pre-determined knowledge of the thermal properties of the materials involved. Data may be obtained by resorting to curve-fitting techniques or extrapolation to substitute for any lack of suitable data.

2.5 LATENT HEAT OF FUSION

One distinct feature which distinguishes the solidification problem from many other heat transfer problems is the release of latent heat of fusion during the phase change. For a steel-casting, this is caused by the change in the free energy associated with the transformation of austenite to ferrite, resulting in the liberation of energy. Most engineering alloys solidify over a range of temperatures rather than at a discrete melting point. Thus, the latent heat of fusion is released over the range of temperatures from the liquidus to the solidus.

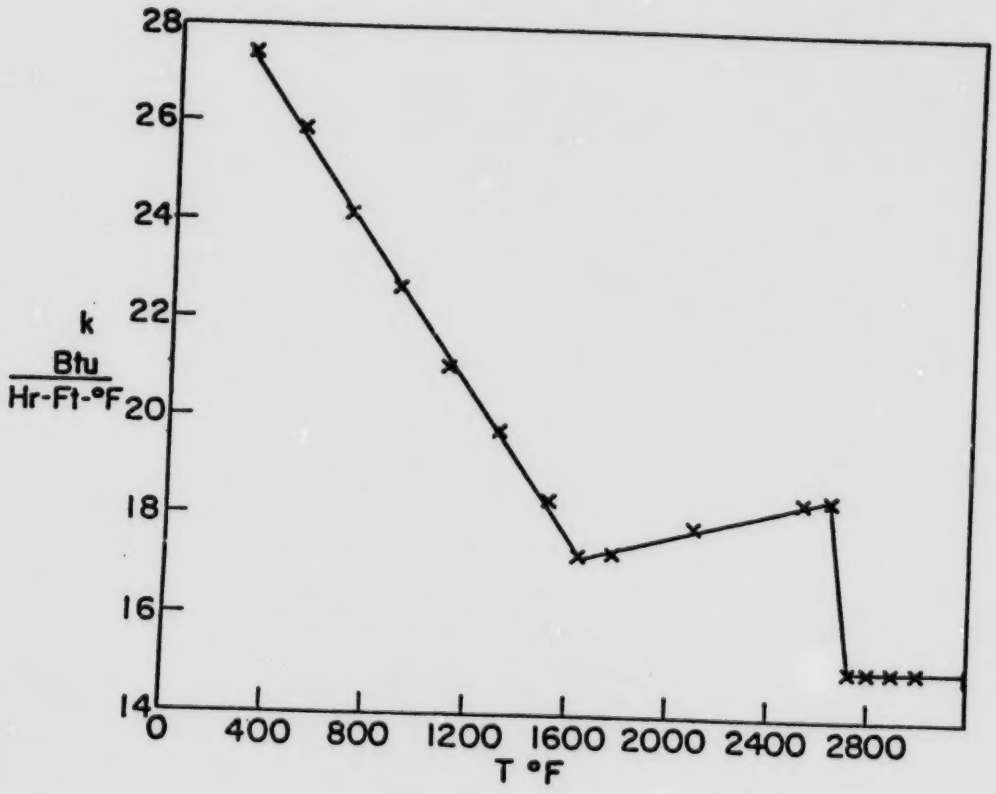


Fig. 2-1 Correlations for the thermal conductivity of Low-Carbon Steel [17].

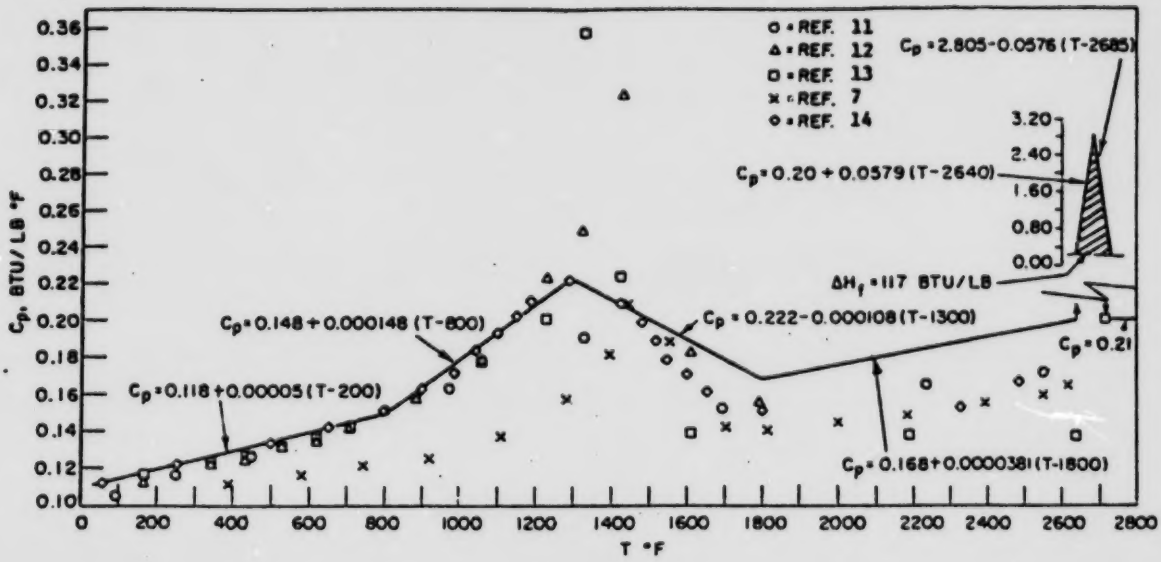


Fig. 2-2 Selected data and functional relationships for specific heat of Low Carbon Steel, including the latent heat approximation [17].

For the present modelling purposes, two methods are used to account for the latent heat of fusion. The first method has already been discussed briefly in the preceding section. In this method, the specific heat within the liquidus-solidus temperature range is artificially increased to simulate the evolution of latent heat. The latent heat of fusion is converted into the appropriate units for specific heat and added to the specific heat term over the liquidus-solidus temperature range, so that

$$C^*_p = C_p + \frac{H}{T_L - T_S} \quad \text{for } T_S < T < T_L \qquad (2\text{-}12)$$

This is done so that the increased area under the specific heat versus temperature curve is equal to the latent heat of fusion as illustrated in Figure 2.2. This method known as the modified specific heat method is suitable for alloys which solidify over a relatively wide temperature range. It is essential that the calculated temperature of the region undergoing solidification should, at some time, fall within the liquidus-solidus range. In a situation where the temperatures are dropping quite rapidly, the choice of too big a time-step may cause the temperatures to drop from above the liquidus to below the solidus over one time-step and prevent any release of latent heat. This method is not suited for pure metals, since the probability of the dropping of the temperature from above the melting point to below the melting point over one time-step is too large.

The second method used to account for the release of latent heat is the post-iterative method in which the temperatures are adjusted after each time-step. The latent heat of fusion is converted into an equivalent number of degrees by dividing the latent heat of fusion by the specific heat, i.e.,

$$DT = \frac{\text{Latent heat of fusion}}{\text{specific heat}} \qquad (2\text{-}13)$$

After each time-step, whenever a node temperature falls below the melting point, the difference between the predicted temperature and the melting temperature is noted. If this difference is less than DT, the node temperature is reset to the melting temperature. This adjustment procedure is continued until the total accumulated difference for that particular node is equal to or greater than DT. At that time, the node temperature is allowed to fall in the normal manner. This procedure is particularly suitable for materials with one melting point.

2.6 DISCRETIZATION OF THE STRUCTURE

In the earlier sections, it was mentioned that the continuum under consideration is a cylindrical mold with the wall of finite dimension. Evidently, the structure has a vertical symmetrical axis. It can be constructed of segments as shown in Fig. (2-3). This is referred to as an element. Each one is characterized by (Δr, $\Delta\phi$, Δz), as shown in the Figure (2-3). So the temperature distribution through the continuum at the specific time can be obtained from 3-Dimensional analysis. Because of the axisymmetrical property of the structure, the temperature field across each half-plane passing through the symmetry axis would be the same. In other words, it is not a function of the circumferential coordinate, $\Delta\phi$. Thus each element defined above can be simplified to a 2-Dimensional element (Δr, Δz). Therefore, it suffices to study the temperature distribution across an arbitrary vertical cross-section through the symmetry axis of the structure as shown in Fig. (2-4).

2.7 DEVELOPMENT OF THE FINITE DIFFERENCE EQUATION IN TWO-DIMENSIONAL CYLINDRICAL POLAR COORDINATES

2.7.1 ENERGY BALANCE METHOD

In the previous section, it was suggested that the given 3-Dimensional continuun can be analysed in a two-Dimensional cylindrical polar coordinate system. The numerical treatment consists of subdividing the structure into a series of small, discrete two-Dimensional elements.

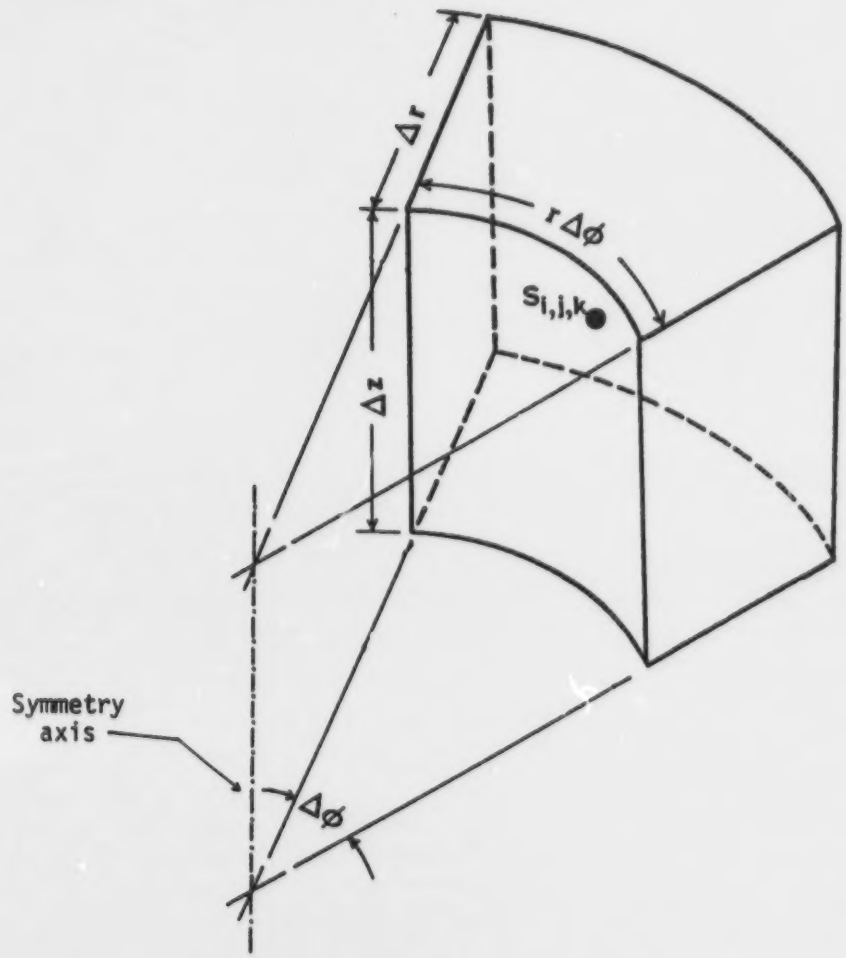


Fig. 2-3 A Three-dimensional element

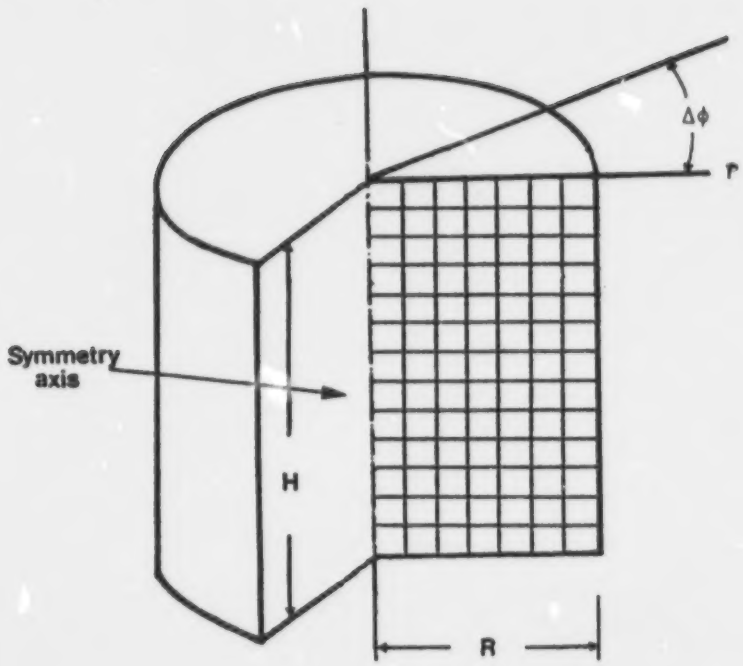


Fig. 2-4 Finite difference grid used in computations.

At the center of each element there is a node with the temperature representive for all points within that particular element. The grid network shown in Figure (2-5) is used with symmetry at the center-line. Note that certain grid lines are assumed to coincide with the centerline, metal and mold boundaries.

The finite difference equation describing the temperature of any particular node can then be obtained by performing the heat balance over the element. The net gain, or loss, of heat between one element and all the surrounding elements goes into raising or lowering the temperature of the element under consideration. The finite difference equations must be derived for each element of the entire set of nodal points, which in general consists of internal nodes, boundary nodes, and interface nodes.

2.7.2 FINITE DIFFERENCE EQUATION FOR AN INTERNAL NODE

The finite difference equation for an internal node can be derived by considering a control volume around the node (i,k) as shown in Figure (2-6) and applying the energy balance to this volume. The control volume associated with the node (i,k) is equal to $(\Delta r_i \Delta z_k)$.

The net heat flow from the surrounding nodes during a small time interval Δt, causes a change in its temperature. Using the ideas described in section (2-2) the finite difference equation is written as

$$K_{i-1,i}\,\Delta z_k\left(r_i-\frac{\Delta r_i}{2}\right)\frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2}+K_{i,i+1}\Delta Z_k\left(r_i+\frac{\Delta r_i}{2}\right)\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_{i+1}+\Delta r_i)/2}$$

$$+K_{k-1,k}\Delta r_i . r_i\frac{(T_{i,k-1}-T_{i,k})}{(\Delta z_{k-1}+\Delta z_k)/2}+K_{k,k+1}\Delta r_i . r_i\frac{(T_{i,k+1}-T_{i,k})}{(\Delta z_{k+1}+\Delta z_k)/2}$$

$$=\rho_{i,k}\,C_{i,k}\,r_i.\Delta r_i\Delta z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}14)$$

where r_i is the radius vector associated with the node (i,k) and $\rho_{i,k}$ and $c_{i,k}$ are the density and specific heat of the body respectively. In the above equation the variables $\Delta r_i, r_i$ have the same significance as Δx_i and x_i in cartesian coordinates respectively.

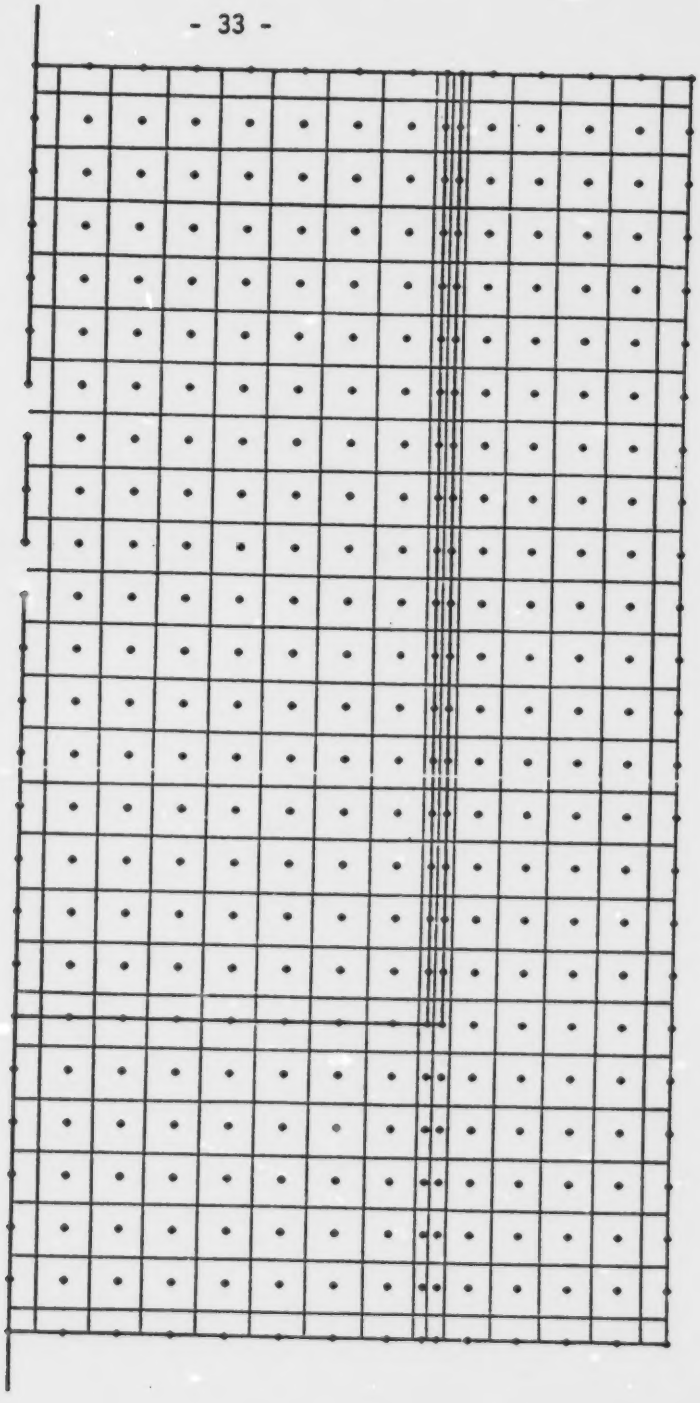


Fig. 2-5 Arrangement of grid points.

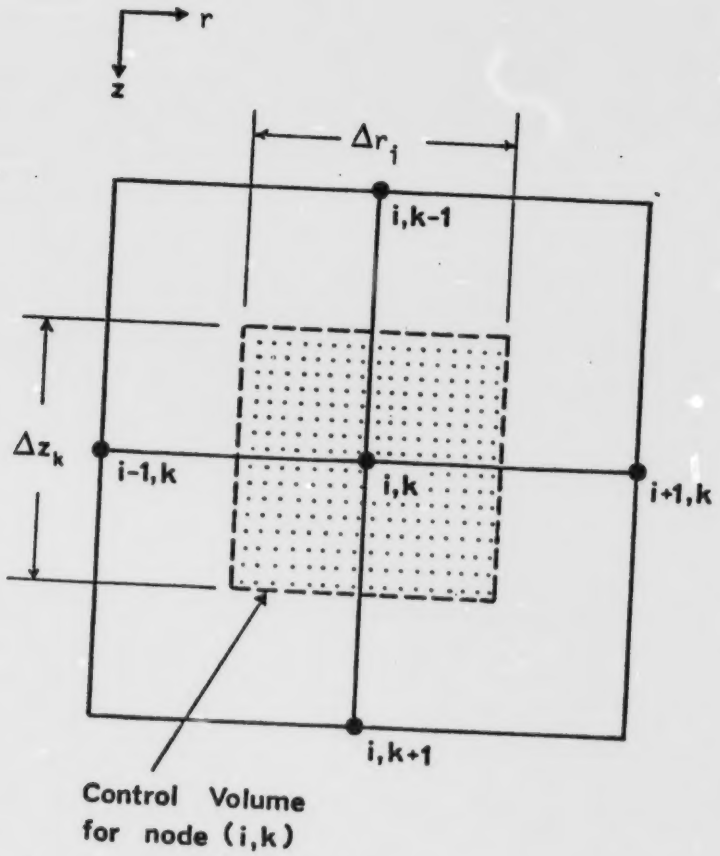


Fig. 2-6 Typical internal node for the two-dimensional situation.

This equation can be derived either by the energy balance method, or from the partial differential equation (P.D.E.) given in section (2-2). In this equation, as well as the others, the coordinate system is chosen as depicted in Figure (2-6).

If the equation (2-14) is divided on both sides by r_i, the efficient configuration of the equation (2-14) for economical computations is obtained as follows;

$$K_{i-1,i}\,\Delta Z_k\left(1-\frac{\Delta r_i}{2r_i}\right)\frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2}+K_{i,i+1}\,\Delta Z_k\left(1+\frac{\Delta r_i}{2r_i}\right)\frac{(T_{i+1k}-T_{i,k})}{(\Delta r_{i+1}+\Delta r_i)/2}$$

$$+K_{k-1,k}\Delta r_i\,\frac{T_{i,k-1}-T_{i,k})}{(\Delta z_{k-1}+\Delta z_k)/2}+K_{k,k+1}\,\Delta r_i\,\frac{(T_{i,k+1}-T_{i,k})}{(\Delta z_{k+1}+\Delta z_k)/2}$$

$$=\rho_{i,k}C_{i,k}\,\Delta r_i\Delta z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t}$$

writing the above equation, for the purpose of comparison in the expanded form, it follows;

$$K_{i-1,i}\,\Delta Z_k\,\frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_{i,k})/2}+K_{i,i+1}\Delta Z_k\,\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_{i+1}+\Delta r_i)/2}$$

$$+K_{k-1,k}\,\Delta r_i\,\frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2}+K_{k,k+1}\Delta r_i\,\frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_{k+1}+\Delta Z_k)/2}$$

$$-\frac{\Delta r_i\Delta Z_k}{2r_i}\left[K_{i-1,i}\,\frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2}-K_{i,i+1}\,\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_{i+1}+\Delta r_i)/2}\right]=\rho_{i,k}C_{i,k}\Delta r_i\Delta Z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t}$$

The similar formula for the cartesian coordinates is;

$$K_{i-1,i}\Delta Z_k\,\frac{(T_{i-1,k}-T_{i,k})}{(\Delta x_{i-1}+\Delta x_i)/2}+K_{i,i+1}\,\Delta Z_k\,\frac{(T_{i+1,k}-T_{i,k})}{(\Delta x_{i+1}+\Delta x_i)/2}$$

$$+K_{k-1,k}\,\Delta x_i\,\frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2}+K_{k,k+1}\,\Delta x_i\,\frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_k+\Delta Z_{k+1})/2}$$

$$=\rho_{i,k}C_{i,k}\,\Delta x_i\,\Delta Z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t}$$

Comparing these two; the left-hand side of the equation in a cylindrical polar coordinate system contains the entire L.H.S. of the corresponding equation in cartesian coordinates and one additional term. The R.H.S. of the two equations are quite similar.

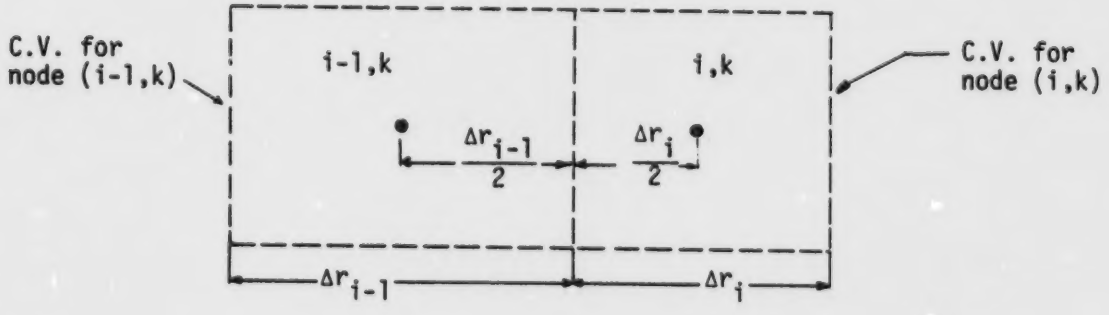
The above comparison indicates that the equation in cartesian coordinates can be obtained from the corresponding equation in cylindrical polar coordinates by eliminating the foregoing additional term in the left-hand side. In practice a binary variable is introduced which can be chosen either as zero or unity. The additional term in cylindrical polar coordinates is multiplied by the above variable. When related to the chosen coordinate system, the variable takes the appropriate value. This type of conversion is being done for the equations of the other type of nodes as well.

Equation (2-14) can also be used for an unequal mesh grid, where Δr_i and Δz_k represent the size of the control volume as shown in Figure 2-6. $K_{i-1,i}$ represents the value of the thermal conductivity pertaining to the control-volume face between the adjacent nodes (i-1,k) and (i,k). The need for evaluating the interface conductivity arises from the fact that the thermal conductivity is temperature-dependent (Section 2-4), and this can lead to a variation in the conductivity in response to the temperature distribution. The control volume surrounding the node (i,k) is considered to be filled with a material of uniform conductivity K_i evaluated at $T_{i,k}$, and the one surrounding node (i-1,k) with a conductivity K_{i-1} evaluated at $T_{i-1,k}$. $K_{i-1,i}$ is taken as the harmonic mean of K_i and K_{i-1}, i.e.

$$K_{i-1,i} = \left[\frac{1-f^r_{i-1,i}}{K_i} + \frac{f^r_{i-1,i}}{K_{i-1}} \right]^{-1} \qquad (2\text{-}15)$$

$$K_{k-1,k} = \left[\frac{1-f^z_{k-1,k}}{K_k} + \frac{f^z_{k-1,k}}{K_{k-1}} \right]^{-1}$$

where $f^r_{i-1,i}$, $f^z_{k-1,k}$ are the interpolation factors which are the ratios defined in terms of the distances as shown,



$$f^{r}_{i-1,i} \equiv \frac{\frac{\Delta r_i}{2}}{\frac{(\Delta r_{i-1}}{2} + \frac{\Delta r_i)}{2}} \quad , \quad f^{z}_{k-1,k} \equiv \frac{\frac{\Delta Z_k}{2}}{\frac{\Delta Z_{k-1}}{2} + \frac{\Delta Z_k}{2}} \qquad (2\text{-}16)$$

Equation (2-14) is used for all internal points in the metal and the mold with their respective thermal properties.

2.7.3 FDE FOR A BOUNDARY NODE

A sketch of the region surrounding a typical side boundary node (i,k) is shown in Figure 2.7. The control volume for node (i,k) is equal to $(\frac{\Delta ri.}{2}\ \Delta z_k)$. Provision is made for both convective and radiative heat flow from the boundary to the ambience at temperature, T_a. Applying an energy balance to the boundary node (i,k), the general form of the finite difference equation is written as:

$$K^{s}_{i-1,i}(r_i - \frac{\Delta r_i}{2})\Delta Z_k \frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2} + h_s . r_i\ \Delta Z_k\ (T_a - T_{i,k})$$

$$+ K^{s}_{k-1,k} . r_i \frac{\Delta r_i}{2} \frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2} + K^{s}_{k,k+1}\ r_i \frac{\Delta r_i}{2} \frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_{k+1}+\Delta Z_k)/2}$$

$$= \rho^{s}_{i,k}\ C^{s}_{i,k}\ r_i \frac{\Delta r_i}{2}\ \Delta Z_k \frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}17)$$

where h_s is the effective heat transfer coefficient accounting for both convective and radiative heat flow for the side walls of the mold as defined in equation (2-8). The ~~subscript~~ upper S indicates that the thermal properties are evaluated for the mold. The FDE for the top and the bottom boundaries are derived similarly. These are given as

$$K_{i-1,i}(r_i - \frac{\Delta r_i}{2})\ \frac{\Delta Z_k}{2} \frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2} + K_{i,i+1}(r_i + \frac{\Delta r_i}{2})\frac{\Delta Z_k}{2} \frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_i+\Delta r_{i+1})/2}$$

$$+ h_T r_i \Delta r_i (T_a - T_{i,k}) + K_{k,k+1} . r_i\ \Delta r_i \frac{T_{i,k+1}-T_{i,k})}{(\Delta Z_k+\Delta Z_{k+1})/2}$$

$$= \rho_{i,k} C_{i,k} r_i \Delta r_i \frac{\Delta Z_k}{2}\ \frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}18)$$

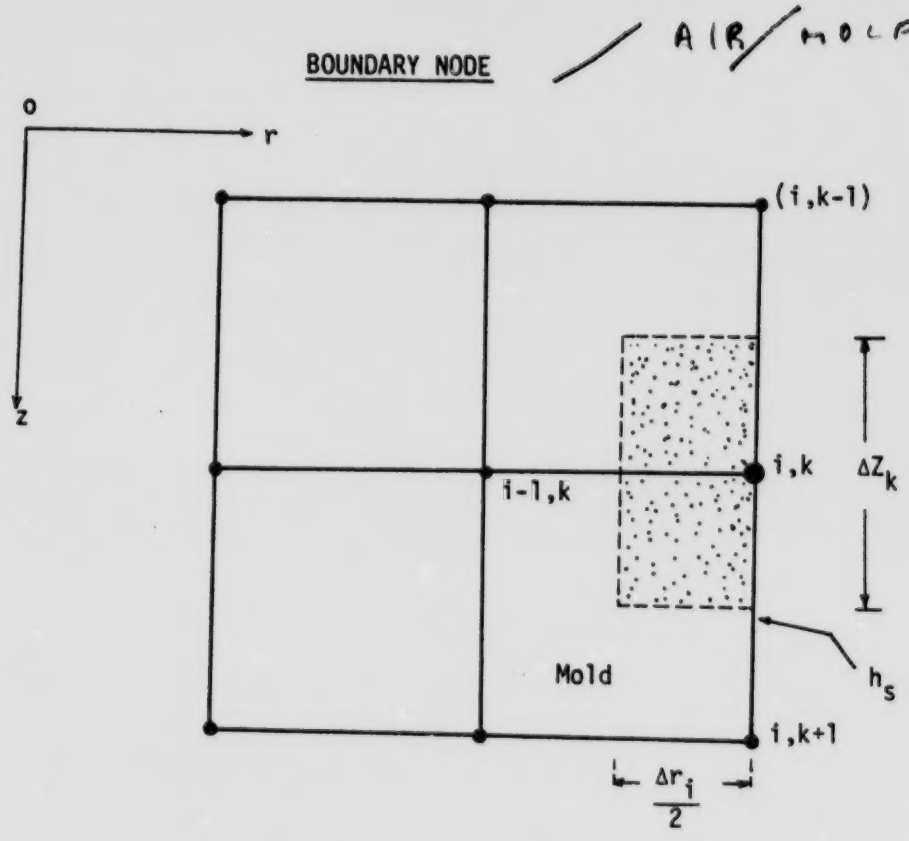


Figure 2-7
Typical boundary node

and

$$K^s_{i-1,i}(r_i - \frac{\Delta r_i}{2})\frac{\Delta Z_k}{2}\frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2} + K^s_{i,i+1}(r_i + \frac{\Delta r_i}{2})\frac{\Delta Z_k}{2}\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_i+\Delta r_{i+1})/2}$$

$$+ K^s_{k-1,k}\cdot r_i \Delta r_i \frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2} + h_b\, r_i \Delta r_i (T_a - T_{i,k})$$

$$= \rho_{i,k} C_{i,k} r_i \Delta r_i \frac{\Delta Z_k}{2}\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}19)$$

with the geometric characters $r_i, \Delta r_i, \Delta Z_k$, and the physical characters $\rho_{i,k}, C_{i,k}$ as given in (2-14), and the superscript S indicates that the thermal properties are evaluated for the mold. In equations (2-18) and (2-19), h_T and h_b are the effective heat transfer coefficients accounting for both convective and radiative heat flows for the top and bottom side of the structure as defined in equation (2-8). The equations relating to the other typical nodes on the boundaries are derived in an analogous way.

2.7.4 FDE FOR A CENTERLINE NODE

A configuration of the area surrounding the centerline nodes is presented in Figure 2-8. Assuming this boundary to be adiabatic, the outward normal temperature gradient is set to zero. This, in turn, leads to setting the conductivity in the radius vector direction to zero. Applying the energy balance method to the typical node (i,K), the following equation is obtained;

$$K_{i,i+1}\Delta Z_k (r_i + \frac{\Delta r_i}{2})\frac{(T_{i+1,\,k}-T_{i,k})}{(\Delta r_{i+1}+\Delta r_i)/2} + K_{k-1,k}\frac{\Delta r_i}{2} r_i \frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2}$$

$$+ K_{k,k+1}\frac{\Delta r_i}{2}\cdot r_i \frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_{k+1}+\Delta Z_k)/2}$$

$$= \tfrac{1}{2}\rho_{i,k} C_{i,k}\, r_i \Delta r_i\, \Delta Z_k \frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}20)$$

This equation can be applied for the centerline nodes using the corresponding conductivity.

The equation is modified for the centerline nodes surrounded by a different medium. For the centerline nodes located at the interface of mold and metal the equation becomes:

CENTERLINE NODE

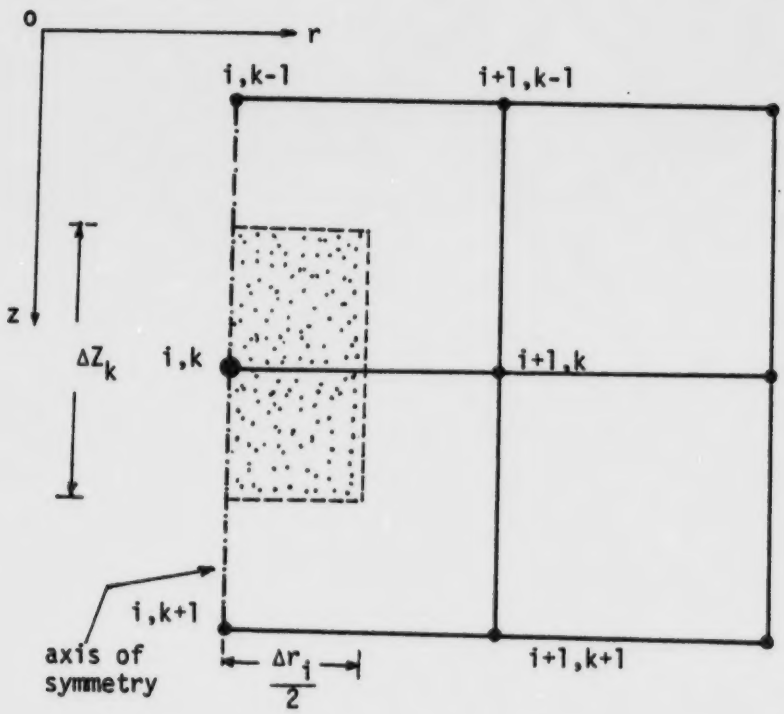


Figure 2-8
Typical centerline node

$$\left[K^m_{i,i+1}+K^s_{i,i+1}\right]\left(r_i+\frac{\Delta r_i}{2}\right)\frac{\Delta Z_k}{2}\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_i+\Delta r_{i+1})/2}$$

$$+K^m_{k-1,k}\,r_i\,\frac{\Delta r_i}{2}\,\frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2}+K^s_{k,k+1}\cdot r_i\,\frac{\Delta r_i}{2}\,\frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_k+\Delta Z_{k+1})/2}$$

$$-\tfrac{1}{4}\left[\rho^m_{i,k}C^m_{i,k}+\rho^s_{i,k}C^s_{i,k}\right]r_i\Delta r_i\Delta Z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}21)$$

In the above formula the control volume is comprised of metal and mold. The right hand side of the equation is evaluated by considering separate contributions from each of the metal and the mold. The equations for the other typical nodes of the centerline are derived similarly. Examples are typical centerline nodes at the top and the bottom boundaries. Referring to Figures (2-9) and (2-10), the corresponding equations are written respectively as;

$$K^m_{i,i+1}\left(r_i+\frac{\Delta r_i}{2}\right)\frac{\Delta Z_k}{2}\,\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_i+\Delta r_{i+1})/2}+h_T\cdot r_i\,\frac{\Delta r_i}{2}\,(T_a-T^{n+1}_{i,k})+$$

$$K^m_{k,k+1}\cdot r_i\,\frac{\Delta r_i}{2}\,\frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_{k+1}+\Delta Z_k)/2}=\tfrac{1}{4}\,\rho^m_{i,k}C^m_{i,k}\,r_i\,\Delta r_i\,\Delta Z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}22)$$

and

$$K^s_{i,i+1}\cdot\left(r_i+\frac{\Delta r_i}{2}\right)\frac{\Delta Z_k}{2}\,\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_i+\Delta r_{i+1})/2}+K^s_{k-1,k}\cdot r_i\,\frac{\Delta r_i}{2}\,\frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2}$$

$$+h_b\,r_i\,\frac{\Delta r_i}{2}\,(T_a-T^{n+1}_{i,k})=\tfrac{1}{4}\,\rho^s_{i,k}C^s_{i,k}\,r_i\,\Delta r_i\Delta Z_k\,\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}23)$$

The sketches of the control volumes for the nodes described by the above equations are shown in the relevant figures.

2.7.5 FDE FOR AN INTERFACE NODE

A sketch of the area surrounding the interfaces node (i,k) between the metal and the mold is shown in Figure (2-11).
Applying the energy balance technique, assuming perfect contact between two faces (before the formation of the air-gap), the equations is written as follows;

$$K^m_{i-1,i}\left(r_i-\frac{\Delta r_i}{2}\right)\Delta Z_k\,\frac{(T_{i-1,k}-T_{i,k})}{(\Delta r_{i-1}+\Delta r_i)/2}+K^s_{i,i+1}\left(r_i+\frac{\Delta r_i}{2}\right)\Delta Z_k\,\frac{(T_{i+1,k}-T_{i,k})}{(\Delta r_{i+1}+\Delta r_i)/2}$$

$$+\left(K^m_{k-1}+K^s_{k-1,k}\right)r_i\,\frac{\Delta r_i}{2}\,\frac{(T_{i,k-1}-T_{i,k})}{(\Delta Z_{k-1}+\Delta Z_k)/2}+(K^m_{k,k+1}+K^s_{k,k+1})r_i\,\frac{\Delta r_i}{2}\,\frac{(T_{i,k+1}-T_{i,k})}{(\Delta Z_{k+1}+\Delta Z_k)/2}$$

$$-\left[\rho^m_{i,k}\,C^m_{i,k}\,r_i\,\frac{\Delta r_i}{2}\,\Delta Z_k+\rho^s_{i,k}C^s_{i,k}r_i\frac{\Delta r_i}{2}\Delta Z_k\right]\frac{(T^{n+1}_{i,k}-T^{n}_{i,k})}{\Delta t} \qquad (2\text{-}24)$$

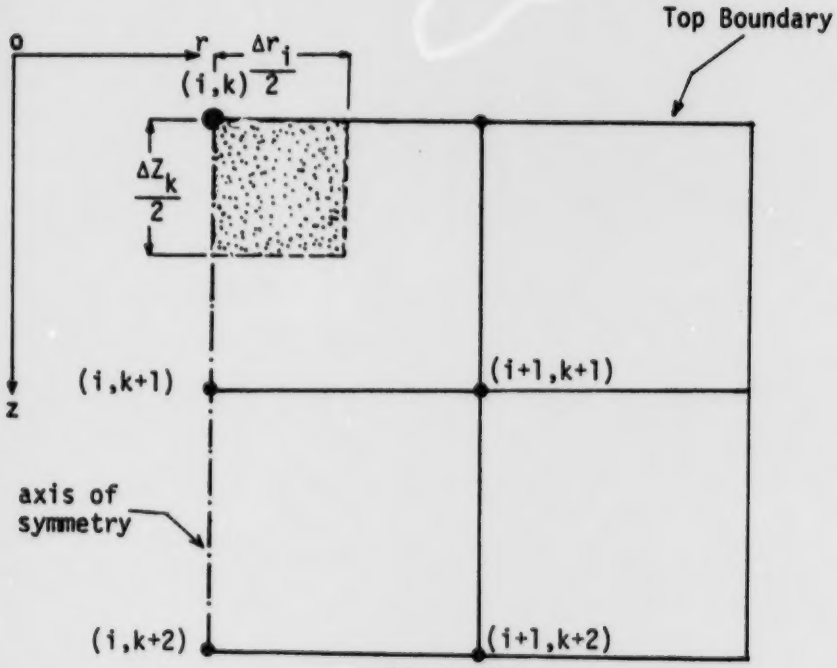


Figure 2-9

Typical centerline node (top corner)

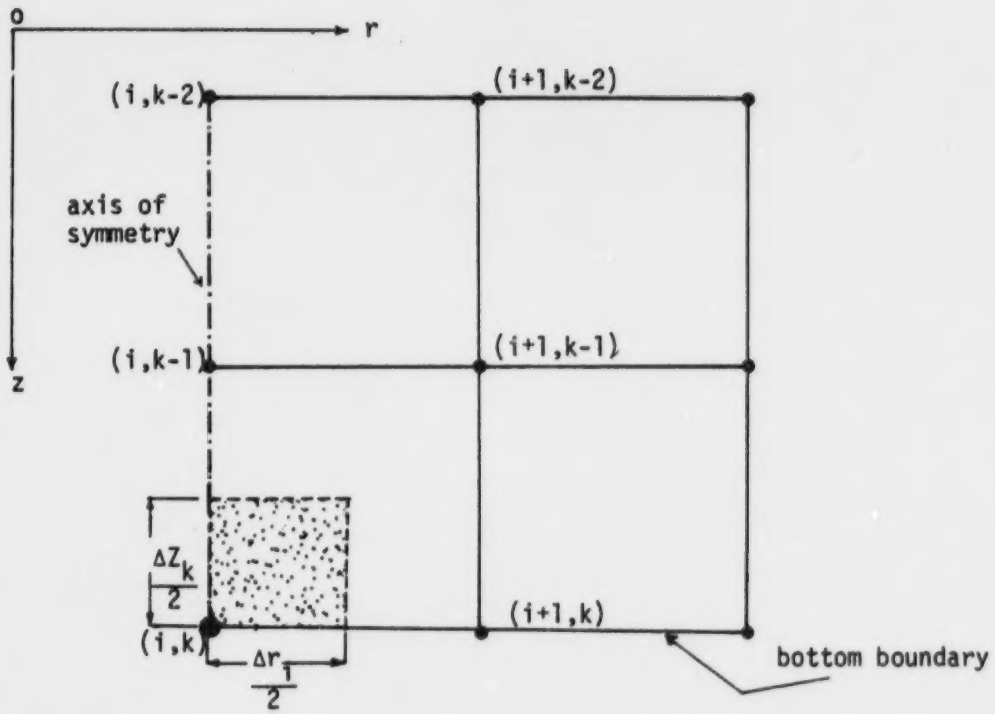


Figure 2-10
Typical centerline node (bottom mold corner)

The superscripts m and s indicate that the thermal properties are evaluated for the metal and the mold respectively. Note that the control volume is comprised of the metal and the mold, and the RHS of equation (2-24) is evaluated by considering separate contributions from each material.

After the formation of the air-gap, the metal and mold are no longer in perfect contact and the derivation of the FDE is handled in the same way as the boundary points with some modifications. Heat transfer is mainly by radiation across the air-gap as discussed in section 2.3. The FDE for node (i,k) located at the face of mold after air-gap formation (breaking apart of ingot and mold) becomes:

$$h_g\, r_i\, \Delta Z_k (T_{i-1,k} - T_{i,k}) + K_{i,i+1} \left(r_i + \frac{\Delta r_i}{2}\right) \Delta Z_k \frac{(T_{i+1,k} - T_{i,k})}{(\Delta r_i + \Delta r_{i+1})/2}$$

$$+ K_{k-1,k}\, r_i \frac{\Delta r_i}{2} \frac{(T_{i,k-1} - T_{i,k})}{(\Delta Z_{k-1} + \Delta Z_k)/2} + K_{k,k+1}\, r_i \frac{\Delta r_i}{2} \frac{(T_{i,k+1} - T_{i,k})}{(\Delta Z_{k+1} + \Delta Z_k)/2}$$

$$- \rho_{i,k} C_{i,k}\, r_i \frac{\Delta r_i}{2} \Delta Z_k \left(\frac{T_{i,k}^{n+1} - T_{i,k}^{n}}{\Delta t}\right) \qquad (2\text{-}25)$$

and the similar equation for the typical node located on the face of metal is expressed as:

$$K_{i-1,i} \left(r_i - \frac{\Delta r_i}{2}\right) \Delta Z_k \frac{(T_{i-1,k} - T_{i,k})}{(\Delta r_{i-1} + \Delta r_i)/2} + h_g\, r_i\, \Delta Z_k (T_{i+1,k} - T_{i,k})$$

$$+ K_{k-1,k}\, r_i \frac{\Delta r_i}{2} \frac{(T_{i,k-1} - T_{i,k})}{(\Delta Z_{k-1} + \Delta Z_k)/2} + K_{k,k+1} \frac{\Delta r_i}{2} \frac{(T_{i,k+1} - T_{i,k})}{(\Delta Z_k + \Delta Z_{k-1})/2}$$

$$- \rho_{i,k} C_{i,k}\, r_i \frac{\Delta r_i}{2} \Delta Z_k \frac{(T_{i,k}^{n+1} - T_{i,k}^{n})}{\Delta t} \qquad (2\text{-}26)$$

where hg is the effective heat transfer coefficient for the air-gap. Similar equations are derived for other typical nodes after air-gap formation.

INTERFACE NODE

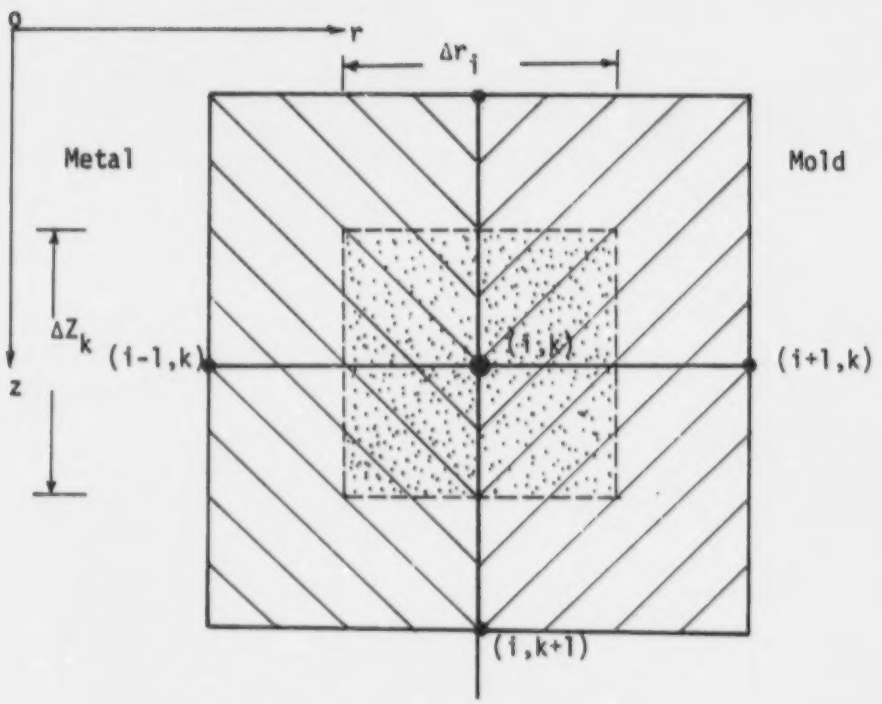


Figure 2-11
Typical interface node

2.7.6 IDENTIFICATION OF FINITE DIFFERENCE EQUATION FOR TYPICAL NODES

In order to choose the appropriate finite difference equation for the typical nodes, a positive integer is associated with each type of node equal to its corresponding finite difference equation number. Figure (2-12) illustrates the numbers for typical nodes in the grid network ranging from 1 to 17. As an example, node number "1" indicates an internal point and the heat balance is as in equation (2-14) with the thermal properties evaluated for the material involved. The node number facilitates the channelling of subsequent calls, in the computer program to the correct FDE for each nodal point. The general procedure for deriving the FDE for all other typical points is similar to the general treatment already discussed in the preceding sections. The computer program contains the full detail, not all of which are given here. A complete listing of the program is attached to the report.

9	5	5	5	5	5	5	5	17	14	5	5	5	5	6
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
7	1	1	1	1	1	1	1	15	11	1	1	1	1	2
8	12	12	12	12	12	12	12	16	13	1	1	1	1	2
7	1	1	1	1	1	1	1	1	1	1	1	1	1	2
7	1	1	1	1	1	1	1	1	1	1	1	1	1	2
7	1	1	1	1	1	1	1	1	1	1	1	1	1	2
7	1	1	1	1	1	1	1	1	1	1	1	1	1	2
7	1	1	1	1	1	1	1	1	1	1	1	1	1	2
10	3	3	3	3	3	3	3	3	3	3	3	3	3	4

Fig. 2-12 Node numbers indicating the type of nodes in the grid arrangement.

CHAPTER 3

NUMERICAL PROCEDURES

3.1 INTRODUCTION

By means of successive application of the energy balance at all nodes in the metal and mold, a series of algebraic equations has been derived which can be solved explicitly or implicitly to obtain the temperature distribution in the system at subsequent time elapses.

The explicit finite-difference method is formulated in terms of the known values. For example, equation (2-14) for all internal points written in explicit form would be:

$$T^{n+1}_{i,k} = T^{n}_{i,k} + \frac{\Delta t}{\rho_{i,k} C_{i,k}} \left[K_{i-1,i} \left(\frac{1}{\Delta r_i} - \frac{1}{2r_i} \right) \frac{(T^{n}_{i-1,k} - T^{n}_{i,k})}{(\Delta r_{i-1} + \Delta r_i)/2} + K_{i,i+1} \left(\frac{1}{\Delta r_i} + \frac{1}{2r_i} \right) \right.$$

$$\left. \frac{(T^{n}_{i+1,k} - T^{n}_{i,k})}{(\Delta r_{i+1} + \Delta r_i)/2} + K_{k-1,k} \frac{1}{\Delta Z_k} \frac{(T^{n}_{i,k-1} - T^{n}_{i,k})}{(\Delta Z_{k-1} + \Delta Z_k)/2} + K_{k,k+1} \frac{1}{\Delta Z_k} \frac{(T^{n}_{i,k+1} - T^{n}_{i,k})}{(\Delta Z_{k+1} + \Delta Z_k)/2} \right] \qquad (3\text{-}1)$$

where the superscript n represents the temperatures evaluated at time = $n.\Delta t$, and the superscript $n + 1$ represents the new temperature at time = $(n + 1).\Delta t$ which is to be calculated. Since the only unknown in this equation is $T^{n+1}_{i,k}$, the new value of the node temperature is computed by substituting the known temperatures into equation (3-1). However, the explicit technique is restricted by the time-step and mesh size to generate a stable solution. Accuracy of the method increases as the time-step and size of the element are decreased. For solidification in a sand mold, for example, the process may take hours, and an accurate solution may be very expensive in computer time.

The implicit method for the solution of FDE may overcome the time-step restriction for stability. The FDE are expressed in terms of the temperatures that are yet to be computed.

The terms in the square bracket in equation (3-1) are evaluated at n + 1 instead of n time-step. The implicit equation now has five unknown temperatures which must be calculated. The application of the FDE to all nodes in the system can generate a series of simultaneous equations which is solved to obtain the new temperature field. Unfortunately, this method requires a solution of a system of simultaneous equations which can be very expensive in terms of computer costs.

To overcome these problems, the model is based on the alternating direction implicit (ADI) method developed by Peaceman and Rachford |24|, since there is no restriction on the size of the time-steps in terms of mathematical stability for the considered simple geometry.

3.2 OUTLINE OF THE ALTERNATING DIRECTION IMPLICIT (ADI) METHOD

In the Peaceman and Rachford scheme, the operation is done in two successive half time-steps, each of duration $\Delta t/2$.

Over the first half time-step, the FDE are expressed implicitly in the x-direction and explicitly in the z-direction. In the successive second half time-step, the procedure is reversed, that is, the equations are made explicit in the x-direction and implicit in the z-direction. Over the two successive half time-steps, the approximation by the ADI method to equation (2-14) are:

$$\rho_{i,k} C_{i,k}\, r_i\, \Delta r_i \Delta Z_k \frac{(T^{*}_{i,k} - T^{n}_{i,k})}{(\Delta t)/2} = K_{i-1,i}\left(r_i - \frac{\Delta r_i}{2}\right) \Delta Z_k \frac{(T^{*}_{i-1,k} - T^{*}_{i,k})}{(\Delta r_{i-1} + \Delta r_i)/2}$$

$$+ K_{i,i+1}\left(r_i + \frac{\Delta r_i}{2}\right) \Delta Z_k \frac{(T^{*}_{i+1,k} - T^{*}_{i,k})}{(\Delta r_{i+1} + \Delta r_i)/2} + K_{k-1,k}\, r_i\, \Delta r_i \frac{(T^{n}_{i,k-1} - T^{n}_{i,k})}{(\Delta Z_{k-1} + \Delta Z_k)/2}$$

$$+ K_{k,k+1}\, r_i\, \Delta r_i \frac{(T^{n}_{i,k+1} - T^{n}_{i,k})}{(\Delta Z_{k+1} + \Delta Z_k)/2} \qquad (3\text{-}2)$$

and

$$\rho_{i,k} C_{i,k}\, r_i\, \Delta r_i \Delta Z_k \frac{(T^{n+1}_{i,k} - T^{*}_{i,k})}{(\Delta t)/2} = K_{i-1,i}\left(r_i - \frac{\Delta r_i}{2}\right) \Delta Z_k \frac{(T^{*}_{i-1,k} - T^{*}_{i,k})}{(\Delta r_{i-1} + \Delta r_i)/2}$$

$$+ K_{i,i+1} \left(r_i + \frac{\Delta r_i}{2}\right) \Delta Z_k \frac{(T^*_{i+1,k} - T^*_{i,k})}{(\Delta r_{i+1} + \Delta r_i)/2} + K_{k-1,k}\, r_i \Delta r_i \frac{(T^{n+1}_{i,k-1} - T^{n+1}_{i,k})}{(\Delta Z_{k-1} + \Delta Z_k)/2}$$

$$+ K_{k,k+1}\, r_i\, \Delta r_i \frac{(T^{n+1}_{i,k+1} - T^{n+1}_{i,k})}{(\Delta Z_{k+1} + \Delta Z_k)/2} \qquad (3\text{-}3)$$

In equation 3-2, over the first half time-step, the T* are unknown and T^n are known from the preceding time-step or initial conditions. Over the second half time-step, the T* in equation 3-3 are now known and the T^{n+1} are unknown. Hence, the successive applications of equation 3-2 followed by equation 3-3 gives the new temperature field after an elapse of time Δt.

Equations 3-2 and 3-3 both contain only three unknowns each. Essentially, by considering a whole row or column of points at once, a system of simultaneous equations for the whole row or column of unknown temperatures is generated. Each equation contains exactly three unknowns except the first and the last equations which contain two unknowns, since the boundary conditions are known or specified as discussed in Section 2.3. This results in a set of equations with tridiagonal coefficients as follows:

$$\begin{aligned}
d_1T_1 + c_1T_2 &= b_1 \\
a_2T_1 + d_2T_2 + c_2T_3 &= b_2 \\
a_3T_2 + d_3T_3 + c_3T_4 &= b_3 \\
a_4T_3 + d_4T_4 + c_4T_5 &= b_4 \\
&\vdots \\
a_nT_{n-1} + d_nT_n &= b_n
\end{aligned} \qquad (3\text{-}4)$$

This system of tridiagonal equations can then be solved quite readily by the Tridiagonal Matrix Algorithm (TDMA).

The ADI method for a two-dimensional situation can be visualized with reference to Figure 3-1. The equations for the grid points along the chosen row contain the unknown temperatures of the chosen row and the temperatures of the two neighboring rows which are known from their latest values.

Thus, the equations for the chosen line are simply tridiagonal equations which are solved by the TDMA. This procedure is carried out for all rows and followed by a similar treatment for the columns for each time-step.

Numerical studies appear to indicate that this method can be used with advantage for rectangular regions. However, under more general conditions it may not possess the same degree of superiority over the other methods. At present, very little is known about the conditions under which this method is valid, and superior to others, when non-rectangular regions and more general equations are employed |30|.

Evidence exists that the method cannot be considered to be stable for any arbitrary time-step.

Despite these limitations, the method has been chosen as the basis for the present model. Its choice was related to the following features of the present model.

1. rectangular regions are used
2. variation in time-step size is limited
3. the specific equations employed are suitable

3.3 COMPUTER PROGRAM

A computer program was developed to solve a two-dimensional solidification problem in either cartesian coordinates or cylindrical polar coordinates, using the alternative direction implicit (ADI) method as outlined in the preceding section.

The program can be run in cartesian or cylindrical polar coordinates depending on whether the binary variable "CYL" is set to zero or unity, which consequently chooses the corresponding set of finite difference equations for the appropriate system of coordinates. A flow chart of the program is shown in Figure 3-2.

The main program starts by reading the input data and computing some preliminary variables. The initial conditions of the temperatures for the metal and mold are also specified at this stage. A call is then made on the subroutine NODAL, which identifies the type of node for each one of the grid points as shown in Figure 2-12.

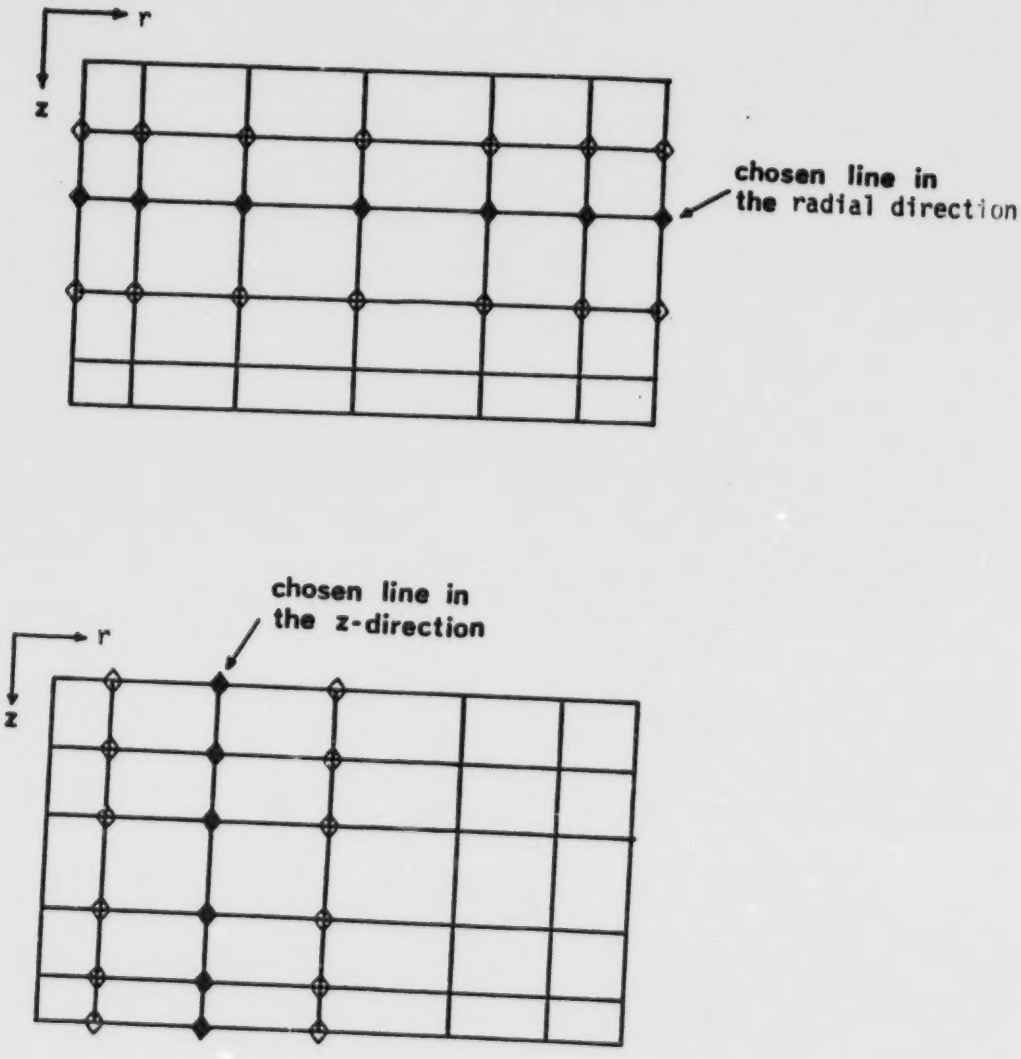


Fig. 3-1 Illustration of line-by-line iteration used in the ADI scheme.

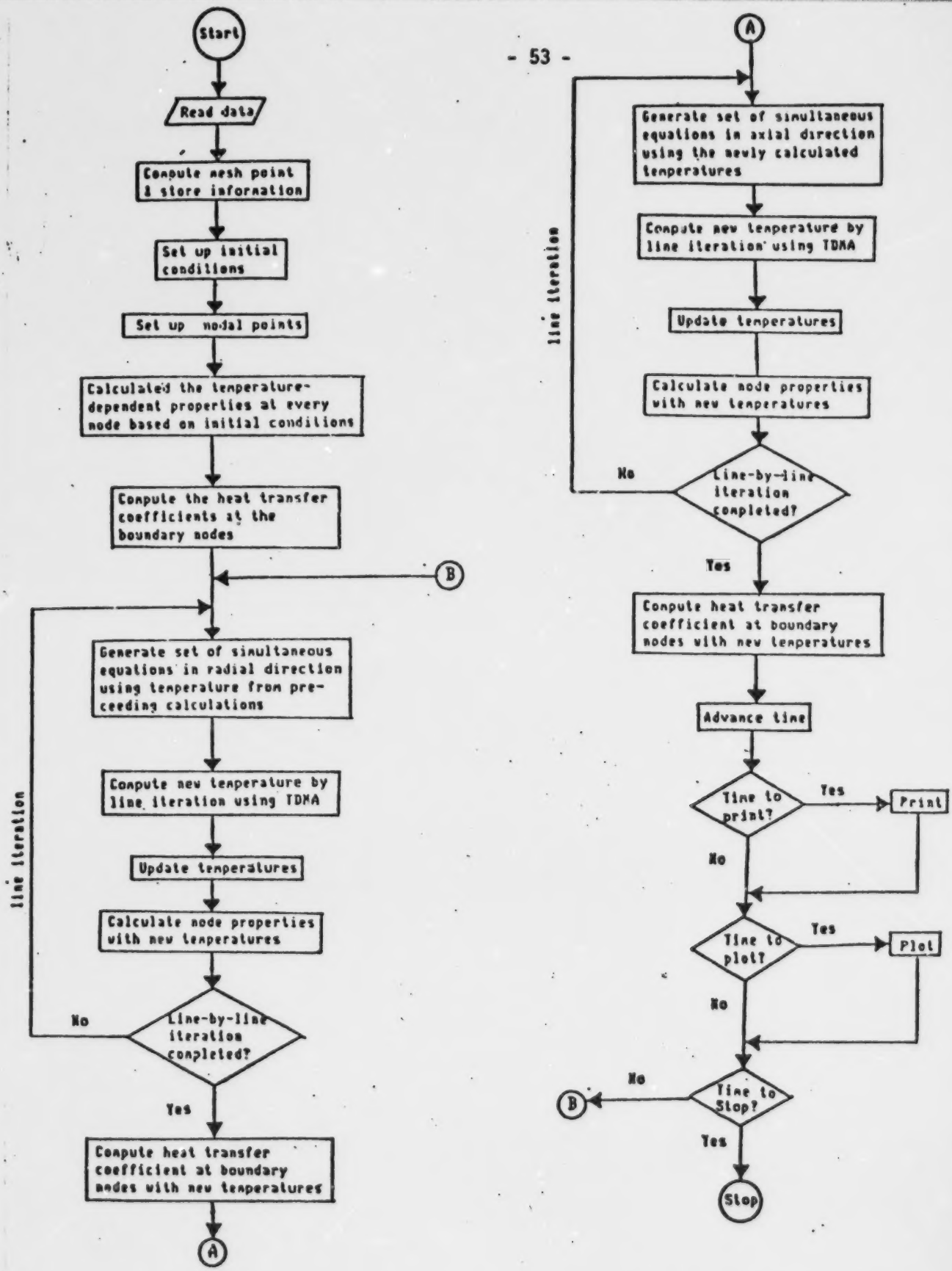


Figure 3.2: Flow chart for calculating solidification in a cast

This facilitates the channelling of the correct finite difference equations to the corresponding grid points. A call is also made on the subroutines to compute the temperature-dependent properties at every node and the heat transfer coefficients at the boundaries based on the given initial conditions.

The rest of the main program is concerned with implementing the ADI method over successive time-steps. Over the first half time-step, the line iterations are solved for the radial direction. Subroutine LINEX determines the set of tridiagonal coefficients for each node along the chosen row, identified earlier by subroutine NODAL. Subroutine TRID is called up to solve the tridiagonal system of simultaneous equations by using TDMA. The chosen line is updated with the newly calculated temperatures, and the thermal properties are adjusted for the new temperatures. This is carried out until all the rows have been completed.

Over the successive second half time-step, subroutine LINEZ is called up to solve the tridiagonal system column by column in a similar way. This completes a whole time-step, and the time is advanced by Δt. At suitable intervals, subroutines PRINT and PLOT are called up to print and plot the resulting outputs. The same procedure is repeated over successive time-steps, until the time exceeds a pre-assigned upper limit TSTOP, at which point computations are discontinued. The thermal properties are determined by using functions RO, SH, and COND to determine the values of the density, specific heat, and conductivity corresponding to the temperature and materials at the different grid points. Latent heat of fusion is accounted for in the specific heat correlations or by the modified specific heat method discussed in Section 2-5. A listing of the FORTRAN program is given in the Appendix.

3.4 GENERAL REMARKS

Generally, by using the ADI method, the solution of the system of simultaneous equations can be obtained quite readily. This method is flexible enough to allow the size of the time-step to vary, without any instability. Though only for simple rectangular regions, the rectangular regions are typical not only for cartesian but also for axisymmetrical cylindrical simple molds.

The treatment of the solidification process involving the release of latent heat, requires some special attention. If the specific heat of a region undergoing solidification is to be increased at the appropriate time, the calculated temperature of that region should, at some time, fall within the liquidus-solidus range. If the time-step is too big, the temperatures may drop from above the liquidus to below the solidus over one time-step. Hence, one does not account for the release of the latent heat as discussed in Section 2-5. To overcome this difficulty, an arbitrary restriction must be imposed upon the size of the time-step to ensure sufficient accuracy. The choice of an appropriate time-step depends on the desired accuracy and the computing costs.

CHAPTER 4

PRELLIMINARY TESTING OF THE MODEL

4.1 INTRODUCTION

During the development of the program, several sample problems were devised for testing. This chapter describes the application of the program for a simple two-dimensional solidification problem defined by the cylindrical polar coordinate system. The reliability of the model is demonstrated by comparing the computed results with those obtained from literature.

4.2 COMPUTATIONAL DETAILS

4.2.1 PROBLEM STATEMENT

The model was applied to simulate the solidification of a ingot in a mold as shown in Figure 4-1 and to determine the temperature distribution in the ingot and mold, the solidification front profile and the solidification time. As it was mentioned earlier, the problem is two-dimensional with symmetry about its centre. An illustration of the two-dimensional vertical plane is shown in Figure 4-1 with the finite difference grids and the boundary conditions.

It was assumed that initially, at time = 0, the entire mold was filled with liquid metal at pouring temperature 1563^{o}C, and no convective transport in the liquid phase was considered. The mold temperature was taken to be uniform at 121^{o}C before computation. Perfect contact was assumed between the ingot and the mold before the formation of the air-gap. The air-gap of finite thickness formed instantaneously after a time elapse of 2 minutes. Heat transfer across the gap was mainly by radiation and conduction due to some contact points. An equivalent heat transfer coefficient was computed as discussed in Chapter 2. The bottom interface between the ingot and mold was assumed to remain in perfect contact during the whole process. Heat was lost to the ambience through the side and bottom surface of the mold. Heat losses at these surfaces were mainly by radiation and convection. The effective heat transfer coefficients for the boundaries was computed, based on the boundary temperatures as discussed. An adiabatic condition was assumed for the top boundary.

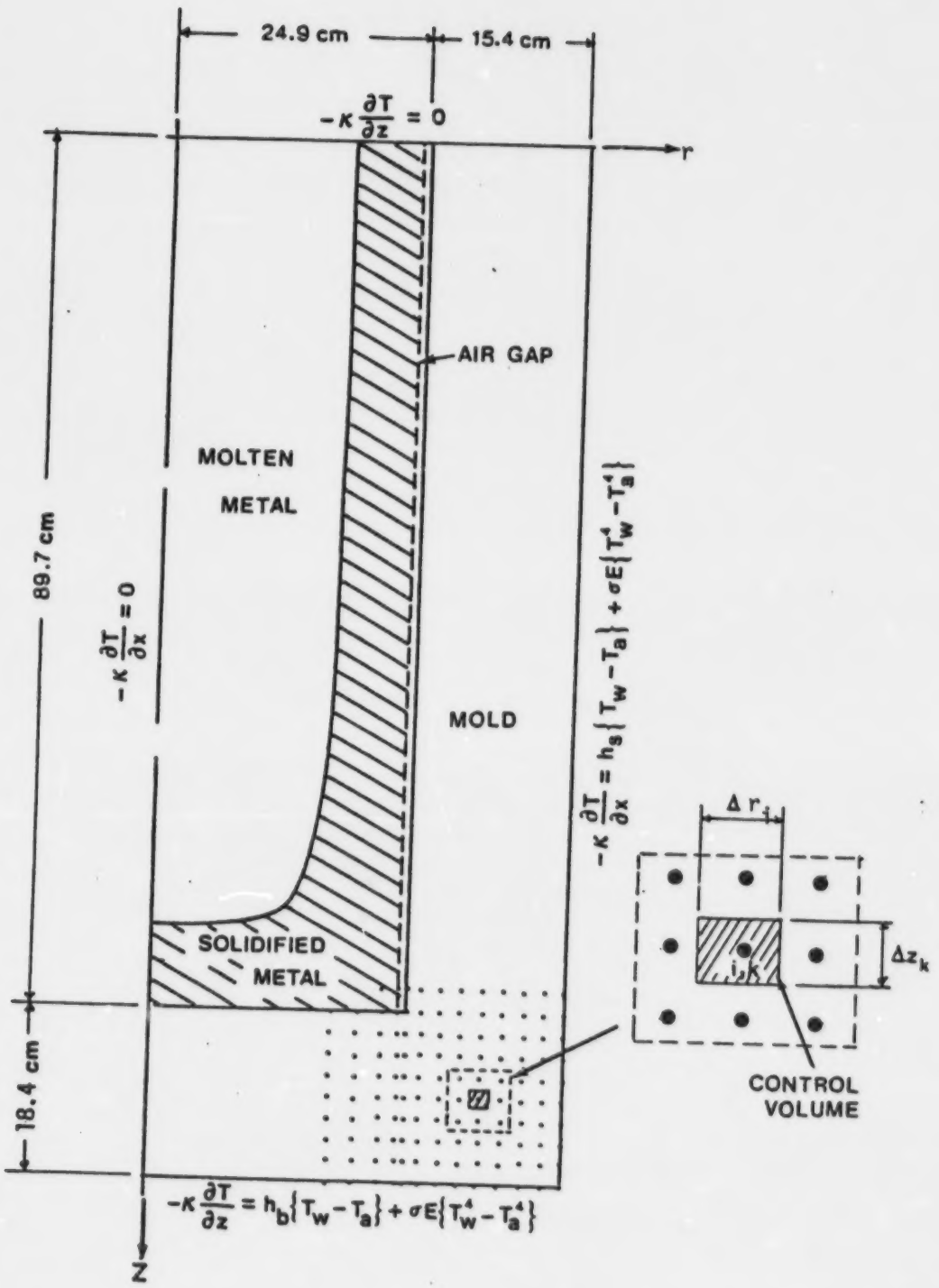


Fig. 4-1 Finite difference grid and boundary conditions.

Linear correlations such as those shown in Figures 2-1 and 2-2 were used for the thermal conductivity and specific heat of the metal and mold. The quantitative expressions used to describe the material properties in this simulation are summarized in Table 4-1. Densities vary only with different materials or different phases of the material and was assumed to be constant over the temperature range. The release of latent heat of fusion was accounted for by the specific heat curve.

4.2.2 SOLUTION PROCEDURES

The shape of the ingot and mold suggests that a rectangular mesh is to be used. Unequal mesh size was used for regions near the air-gap and interface. A total of 960 grid points was used with 20 nodes in the radial-direction and 48 nodes in the z-direction. The finite difference equations and numerical scheme presented in the preceeding chapters were used to compute the temperature distributions in the ingot and mold during the process. A time-step of 10 seconds was used in the computation. The program was allowed to run for 80 minutes of simulated time. The temperature for each grid point was printed at different time intervals. A sample output from the program is shown in Figure 4-2. From these results, the temperature and solidification front profiles were plotted and analyzed. Some sample results are illustrated and discussed in the next section.

4.3 SAMPLE RESULTS AND DISCUSSION

The nature of the solidification pattern is illustrated in Figure 4-3. The location of the solidification front is depicted for different time intervals. Solidification starts from the metal/mold interface and proceeds towards the centre of the ingot. Note that the initial solidification rate is quite fast, but slows down as it approaches the ingot core. The nature of curve is similar to that obtained by Raychaudhuri [27]. A plot of the solidified thickness versus the square root of the time is presented in Figure 4-4. The graph can be approximated by a linear correlation given by

$$x = 2.5625 \sqrt{t} \qquad (4\text{-}1)$$

where x is given in centimeters and t in minutes.

Table 4.1. Input data for solidification program.

Thermal properties of low carbon steel

Density: liquid = 7200.0 kg/m³
solid = 7480.6 kg/m³

Liquidus temperature: 1498.8 °C (2730.0°F)

Solidus temperature: 1448.8 °C (2640.0°F)

Latent heat of fusion: 272.1 kJ/kg

Thermal conductivity (from Fig. 2-1)

0° to 1640°F, K = 28.58-0.00722 (T - 100.0)
1640° to 2640°F, K = 17.3 + 0.0012 (T - 1640.0)
2640° to 2730°F, K = 18.5 - 0.0389 (T - 2640.0)
>2730°F, K is constant at 15.0

(T is in °F and K in BTU/hr - ft - °F)

To convert K to S.I. units, multiply by 1.7307.

Specific heat (from Fig 2-2)

0° to 800°F, C_p = 0.118 + 0.00005 (T - 200.0)
800° to 1300°F, C_p = 0.148 + 0.000148 (T - 800.0)
1300° to 1800°F, C_p = 0.222 - 0.000108 (T - 1300.0)
1800° to 2640°F, C_p = 0.168 + 0.0000381 (T - 1800.0)
2640° to 2685°F, C_p = 0.20 + 0.0579 (T - 2640.0)
2685° to 2730°F, C_p = 2.805 - 0.0576 (T - 2685.0)
>2730°F, C_p is constant at 0.21

(T is in °F and C_p in BTU/lb-°F).

To convert C_p to S.I. units, multiply by 4186.9.

Pouring temperature: 1563.0°C (2845°F)

Mold temperature: 121°C (250°F)

Ambient temperature: 23.8°C (75°F)

Dimensions of ingot and mold: see Fig. 4-1

By extrapolation, the solidification time for the ingot was found to be about 89 minutes. This was compared with the correlation suggested by Flemings |8| for steel solidification against cast-iron mold walls which is of the form

$$x = a\sqrt{t} - b \qquad (4\text{-}2)$$

where the constant a is approximately unity and the constant b is 0.5 when the units are in inches and minutes. Figure 4-4 is a graph derived from plotting the results. Both results agree reasonably well.

In Figure 4-5, the temperature profiles in the ingot are presented for (a) time = 10 minutes and (b) time = 40 minutes after pouring. Heat losses from the ingot to the mold are quite uniform in the transverse direction. A comparison of the two temperature distributions show that initially most parts of the ingot around the centre core are not very much affected by the heat losses to the mold. Most of these regions are still in their molten form and the temperature distributions are only slightly below the pouring temperature. The solidified parts of the ingot lose heat quickly to the relatively cold mold as indicated by the crowding of the isotherms. As time progresses, more metal in the ingot solidifies and the mold gains more heat from the ingot, the isotherms are spread more evenly.

Figure 4-6 shows the temperature profiles in the mold for the same time interval as Figure 4-5. The isotherms at time = 10 minutes are generally at a lower temperature. The mold appears to be gaining more heat from the ingot than losing heat to the ambience. At time = 40 minutes, the isotherms are at a higher temperature since the mold is heated up. This information would be useful for designing molds and also for optimizing the cooling time for the mold before it is used again.

Figure 4-7 presents the temperature gradients in the ingot and mold at time = 10 minutes. A study of this figure and the preceeding two figures confirmed the observation of high heat flow in the solidified region as indicated by the steep temperature gradient. Note the temperature discontinuity at the ingot/mold interface because of the air-gap. At a lower level in the mold, there is no discontinuity. Figure 4-8 illustrates similar results for time = 40 minutes.

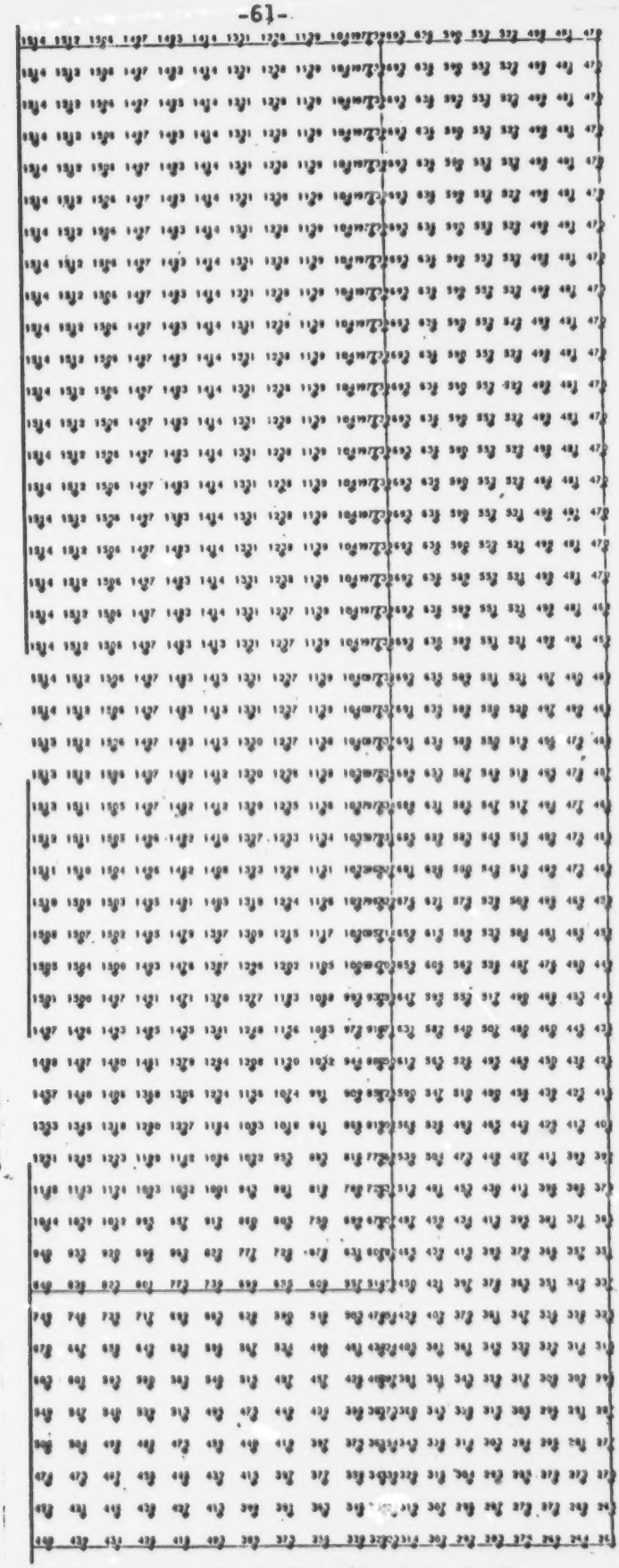


Figure 4-2: Temperature distribution at time = 30 minutes.

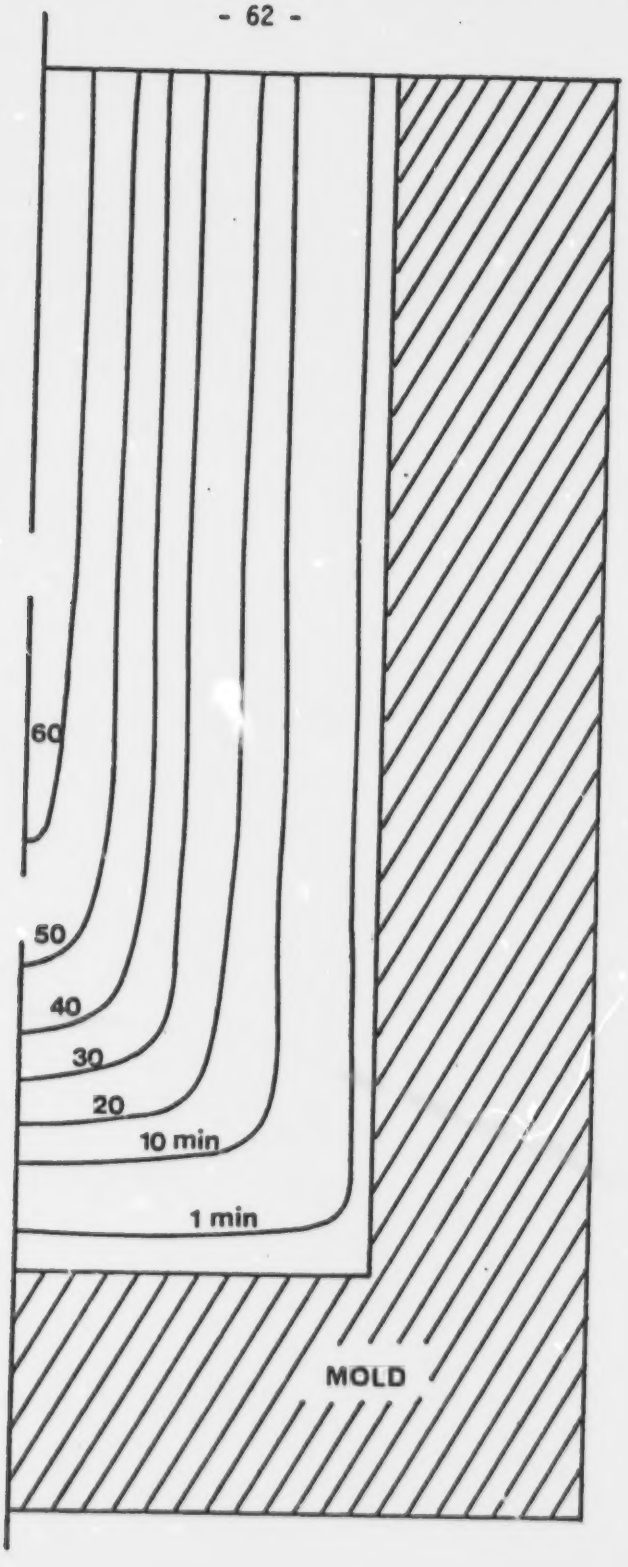


Fig. 4-3 Advancement of solidification front in a cylindrical ingot.

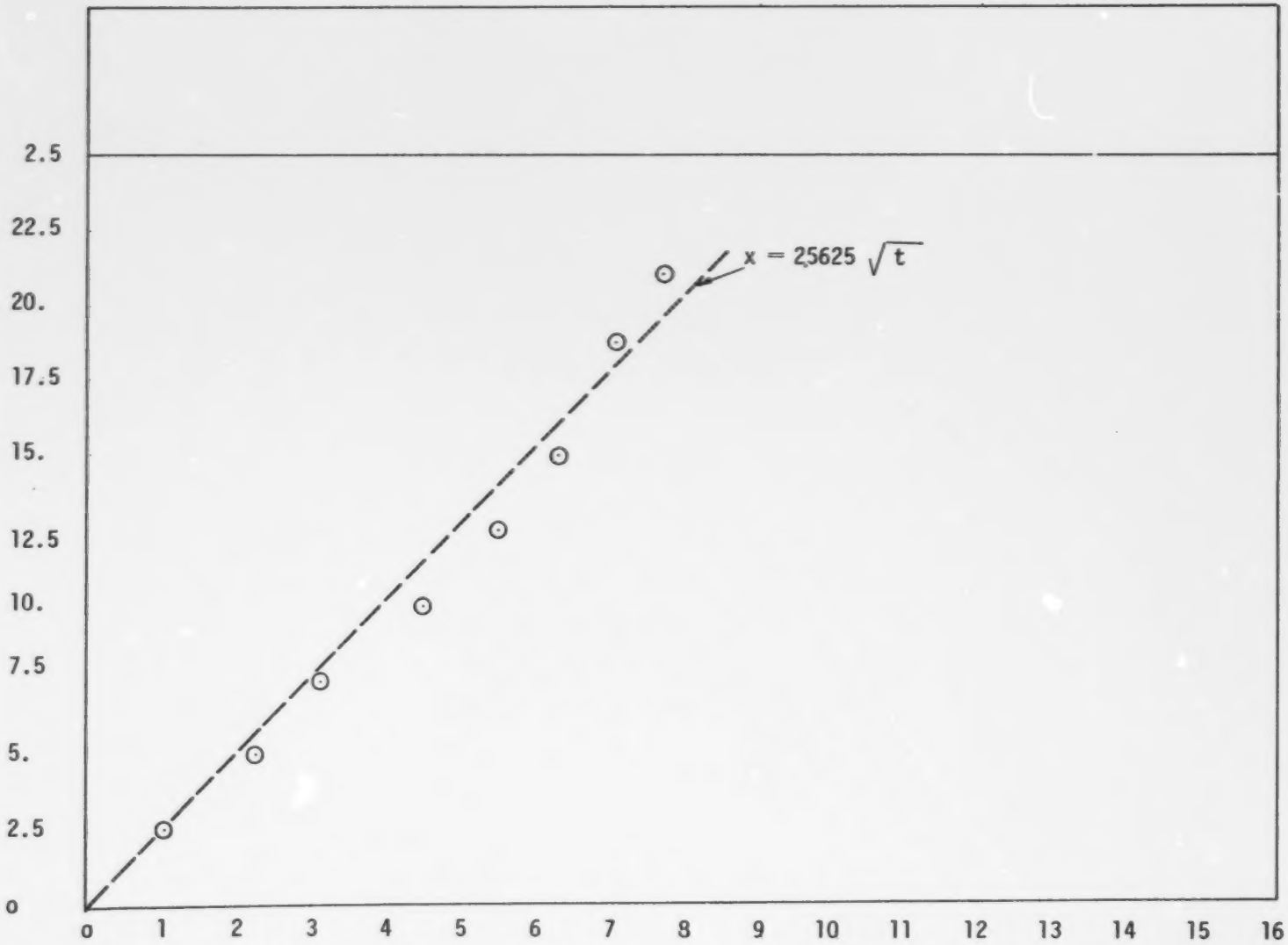


Figure 4-4 Solidified thickness versus square root of time t. (min. ½).

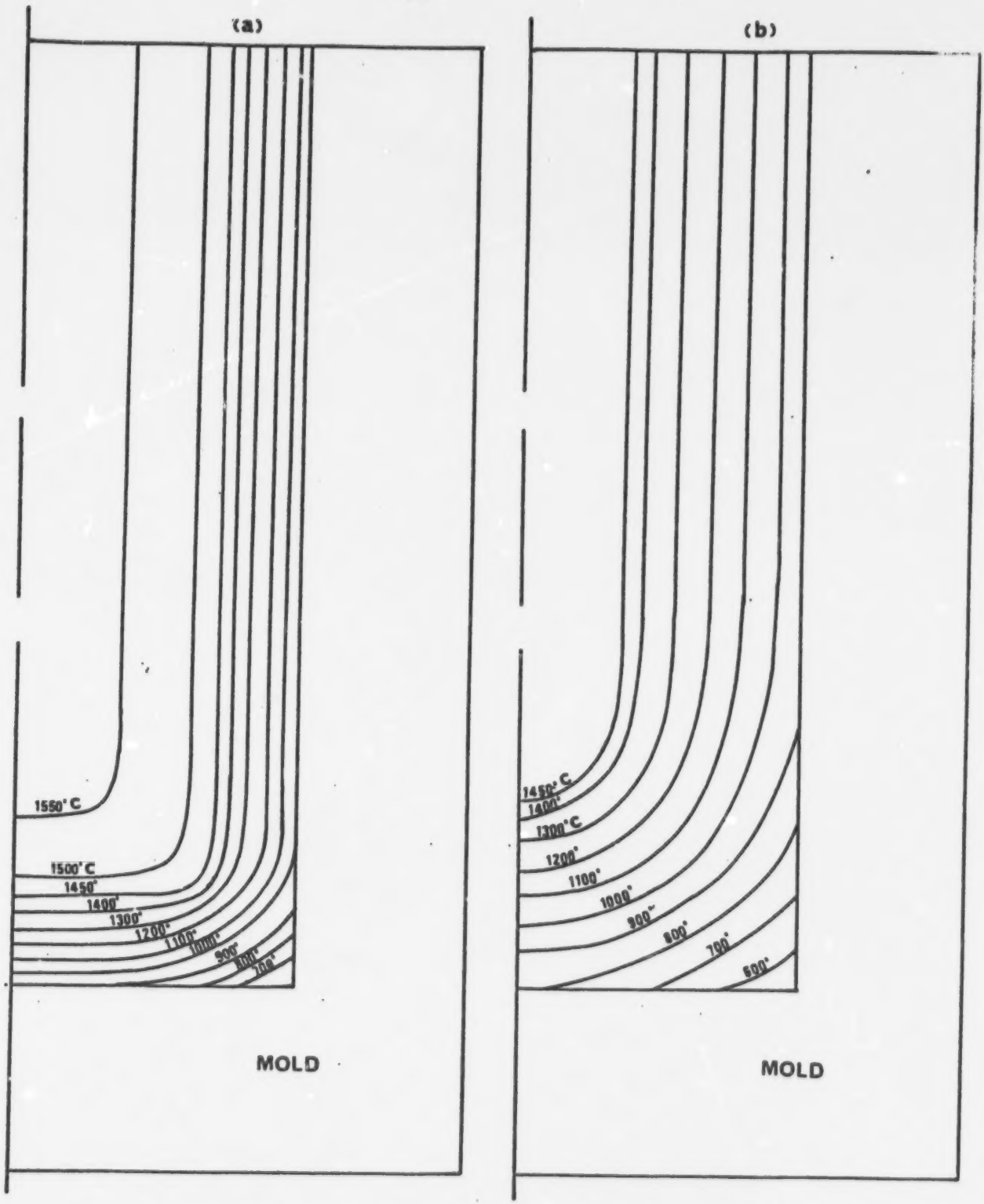


Fig. 4-5 Isotherms in the cylindrical ingot at (a) time = 10.0 min. and (b) time = 40.0 min. after pouring.

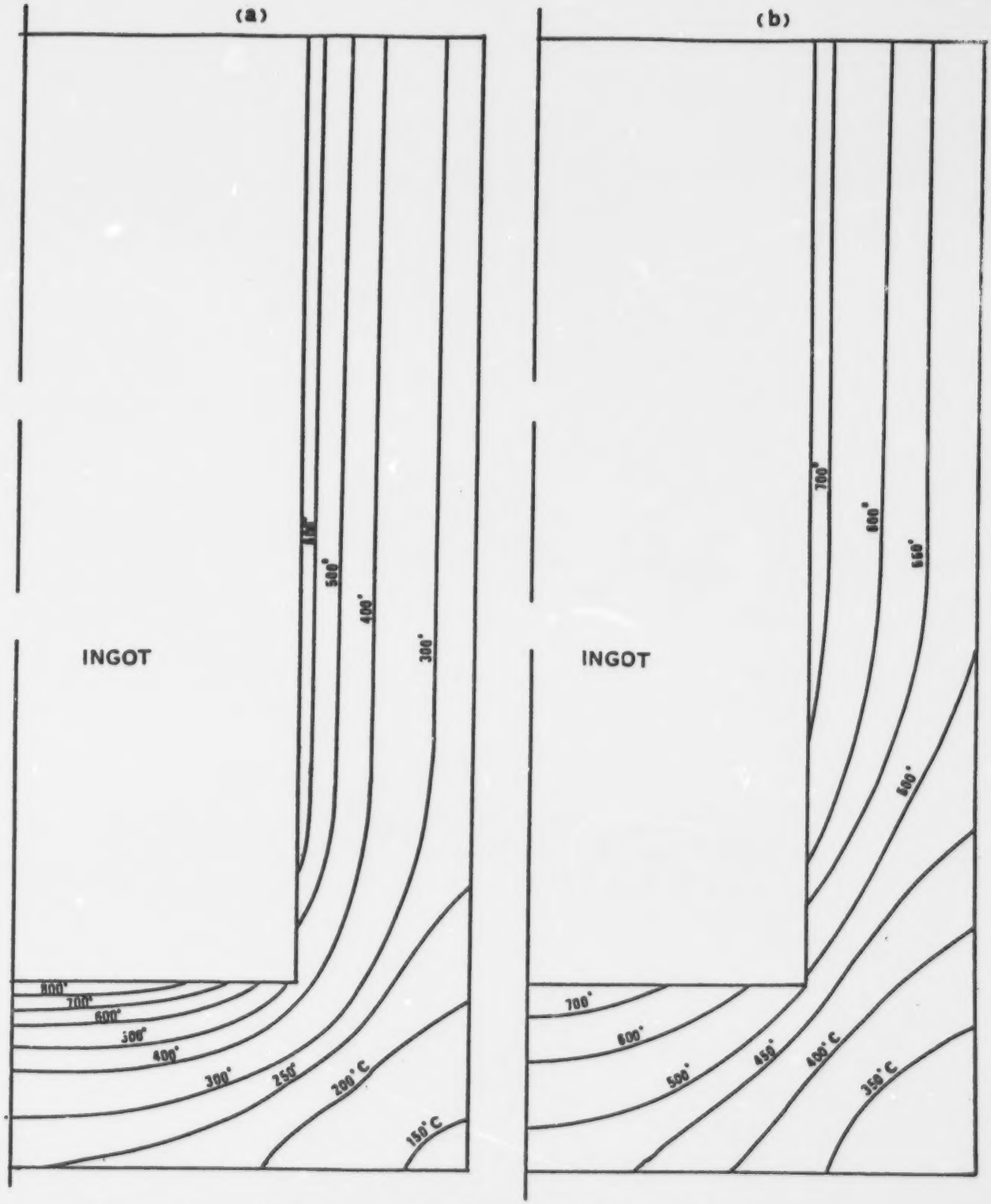


Fig. 4-6 Isotherms in the cylindrical mold at (a) time = 10.0 min. and (b) time = 40.0 min. after pouring.

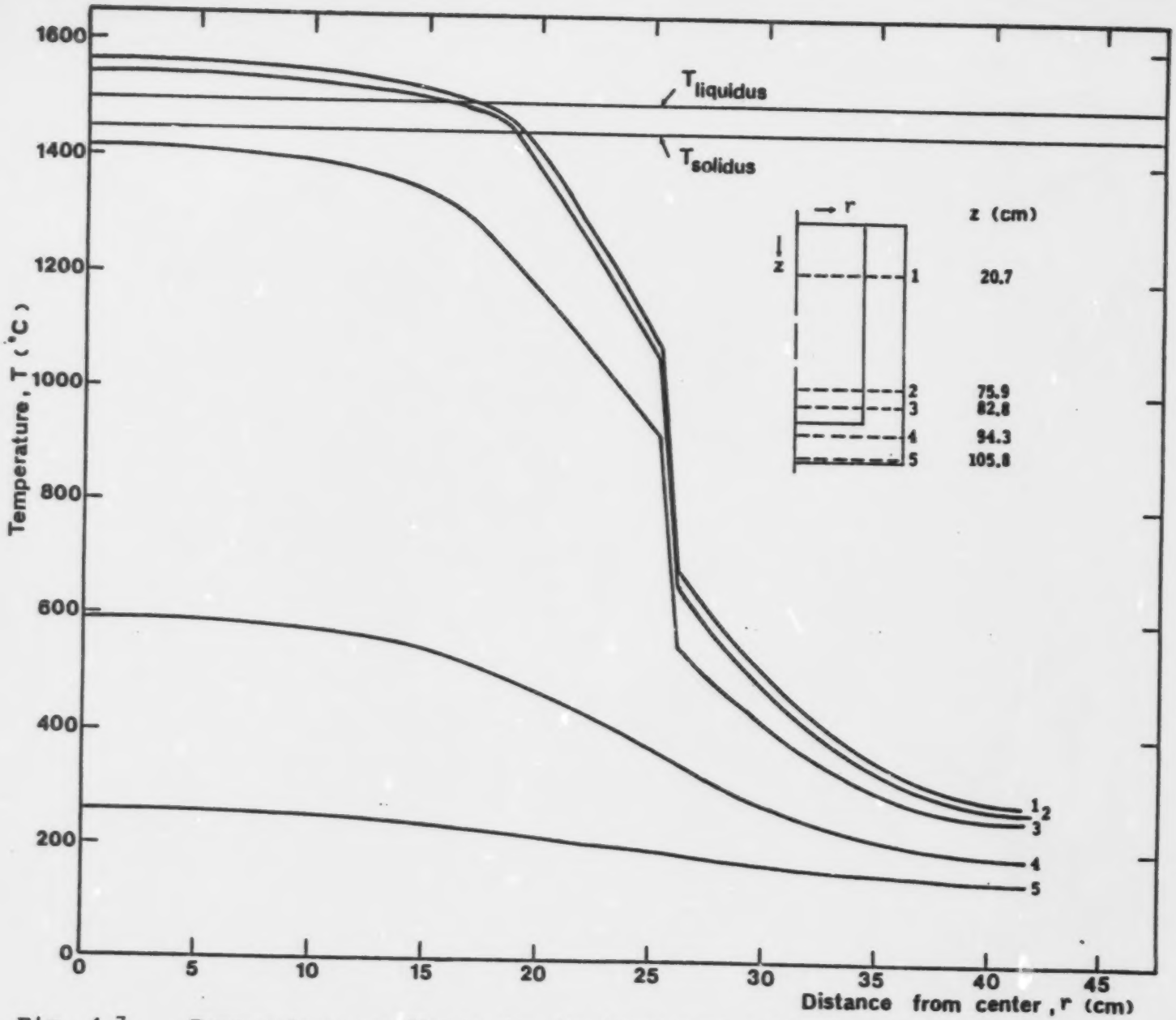


Fig. 4-7 Temperature gradients in the ingot and mold at time = 10.0 min. after pouring.

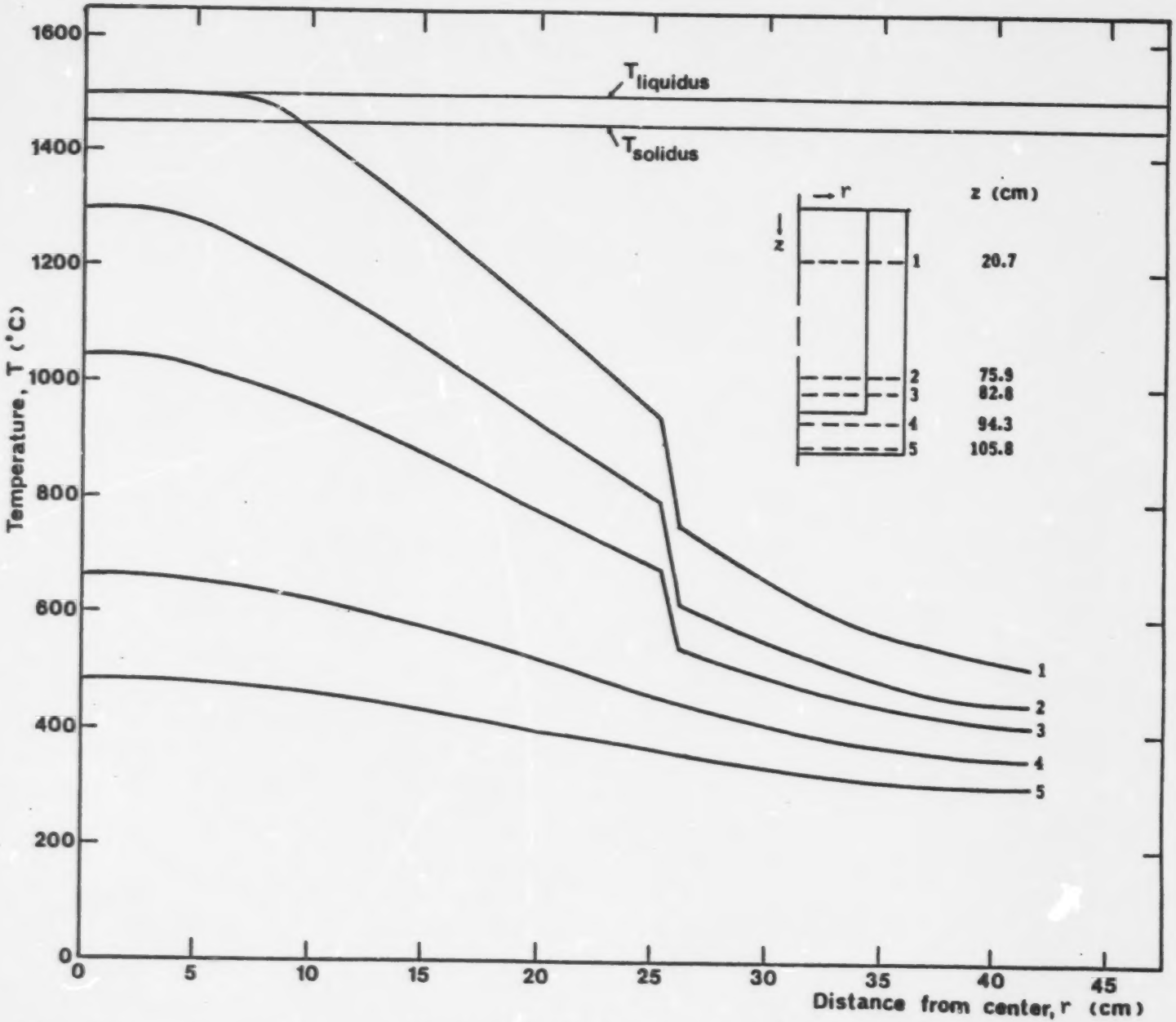


Fig. 4-8 Temperature gradients in the ingot and mold at time = 40.0 min. after pouring.

Making another study with the preceeding figures, the temperatures in the mold increased while that in the ingot decreased as observed earlier. Because of this, the temperature drop across the interface decreased.

The cooling curves at the centre of the ingot are shown in Figure 4-9. The temperatures at the centre core are plotted against time at different heights of the ingot. Since the top boundary is adiabatic, heat losses near the top are very slow. The temperature near the bottom of the ingot dropped quite significantly over the first 10 minutes and then levelled off. Analyzing this figure with Figure 4-3, one notices that the drop in the temperatures is more significant after the metal surrounding the node point has solidified.

It is interesting also to look at the temperature-time variation in the ingot and mold at a height 66.7 cm from the top shown in Figure 4-10. Lines 1 and 2 show the typical cooling curves in the ingot similar to those discussed in Figure 4-9. The temperature variation at the outer surface of the ingot is represented by line 3. The effect of the air-gap is quite clear in the figure. Before the formation of the gap, the temperature dropped rapidly from its pouring temperature since perfect contact was assumed. At time = 2 minutes, as observed by Raychaudhuri |27|, the temperature rises again because of considerable resistance to the heat flow caused by the gap. As time progresses, more heat is lost from the ingot and the temperature falls again. The temperature variation at the inner surface of the mold is also interesting. This node, before the gap formation is the interface node, and the temperature at time = 0 computed from equation 2-9 is initially 833^{o}C. The temperature rises slightly and drops after the gap has formed. It then rises when the mold is heated up and begins to drop after a while. Lines 5 and 6 show the temperature variation at the middle and outer surface of the mold. Initially, there is a slight drop at the outer surface because heat is lost to the ambience but heat gained from the ingot is much greater and it begins to rise to a maximum and cool down.

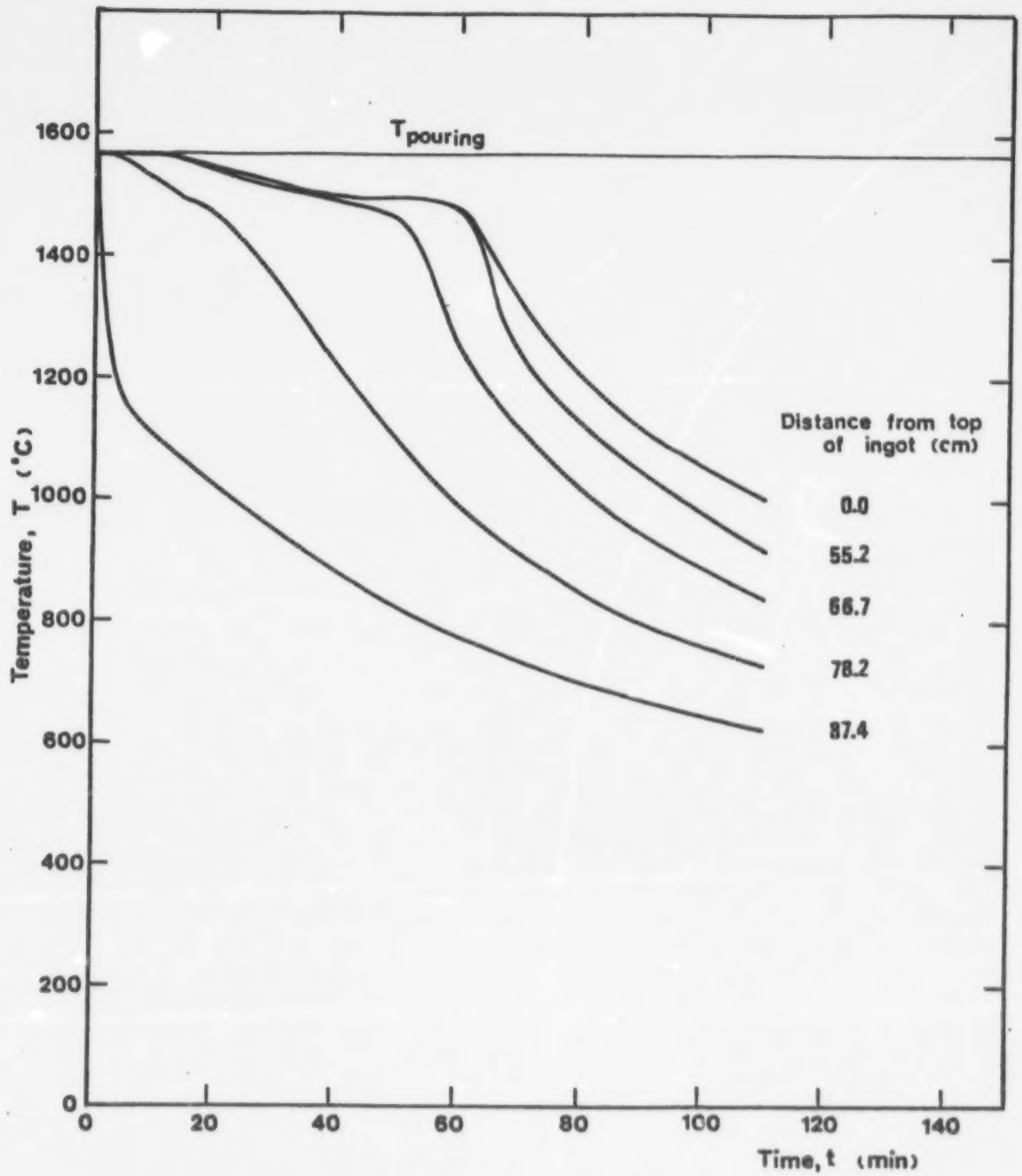


Fig. 4-9 Cooling curves at the center of ingot.

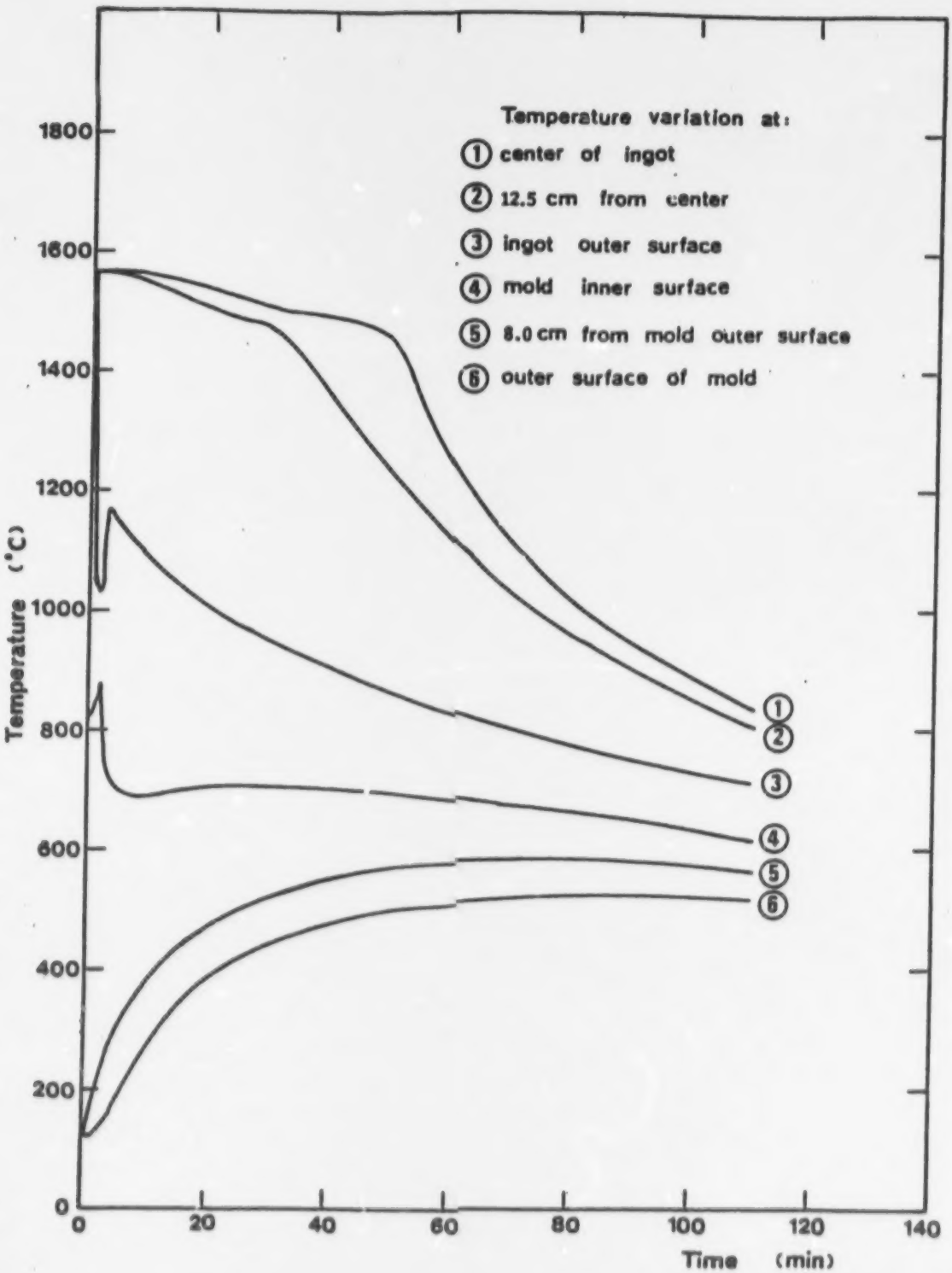


Fig. 4-10 Ingot and mold temperature-time variation at 66.7 cm from top of ingot.

The effective heat transfer coefficients for the boundaries are computed at tne beginning of each time-step based on the temperature at the boundary nodes. Figures 4-11 to 4-13 present the computed heat transfer coefficients for the side, bottom and interface respectively. Figures 4-11 and 4-12 display similar pattern. The values of the heat transfer coefficient increase to a maximum as the temperatures of the nodes rises and starts to decrease after. The nodes with higher temperature appear to have a higher heat transfer coefficient. The variation of heat transfer coefficient at the interface is depicted by Figure (4-13).

The work in the testing of the program for simple geometries is still in progress. Many other variations for testing the model can be done by varying the mold thickness, size, or initial temperature; or varying the size of the ingot, by changing to different metals and molds. Information from these results can be used to improve the qualitative understanding of the influence of the various parameters in the solidification process and which, in turn, can be used for process control and design. The main limitation of the model is the lack of useful data for the thermal properties of the materials involved which can severely affect the accuracy of the results.

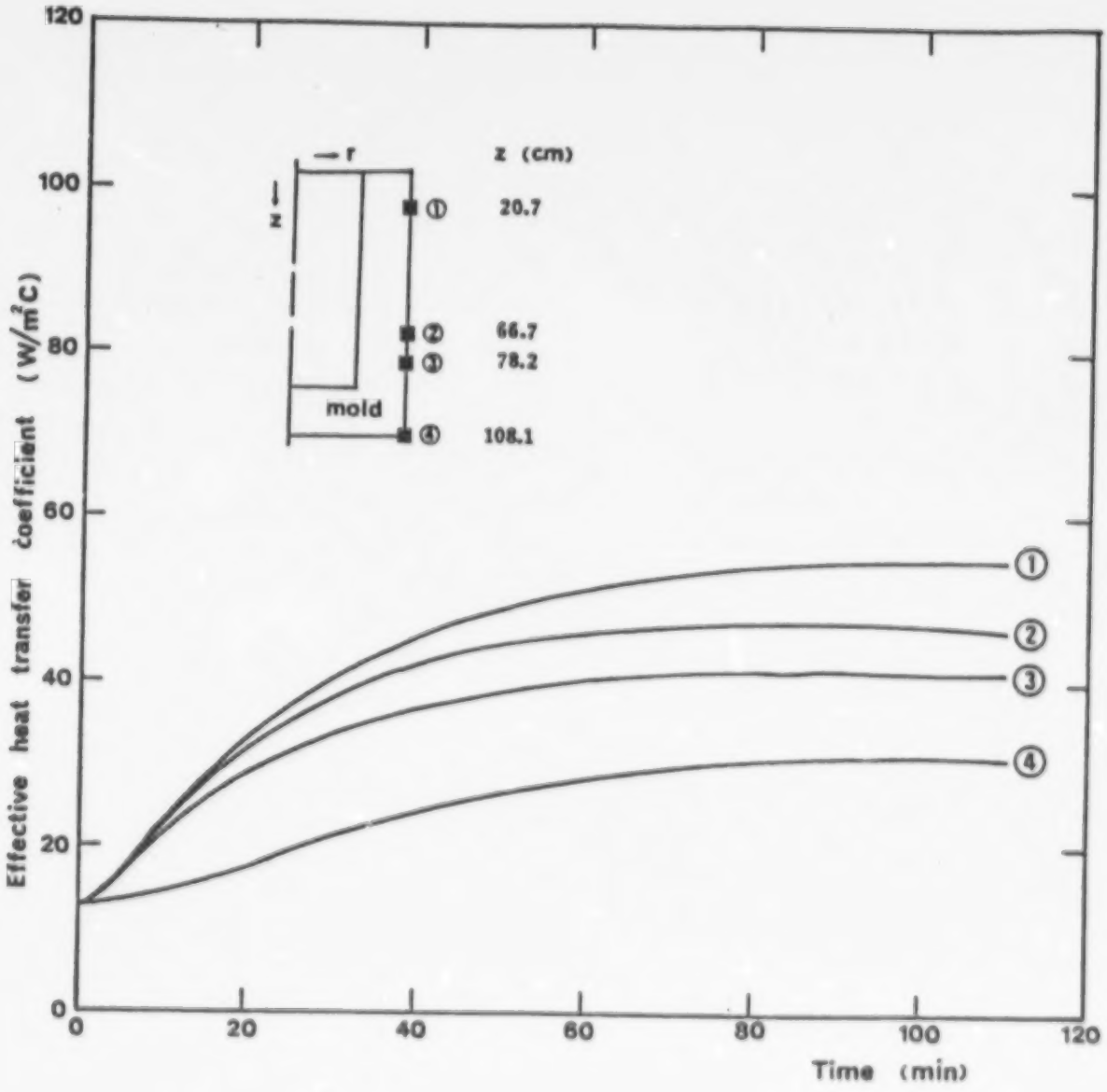


Fig. 4-11 Variation of effective heat transfer coefficient at side outer surface of mold.

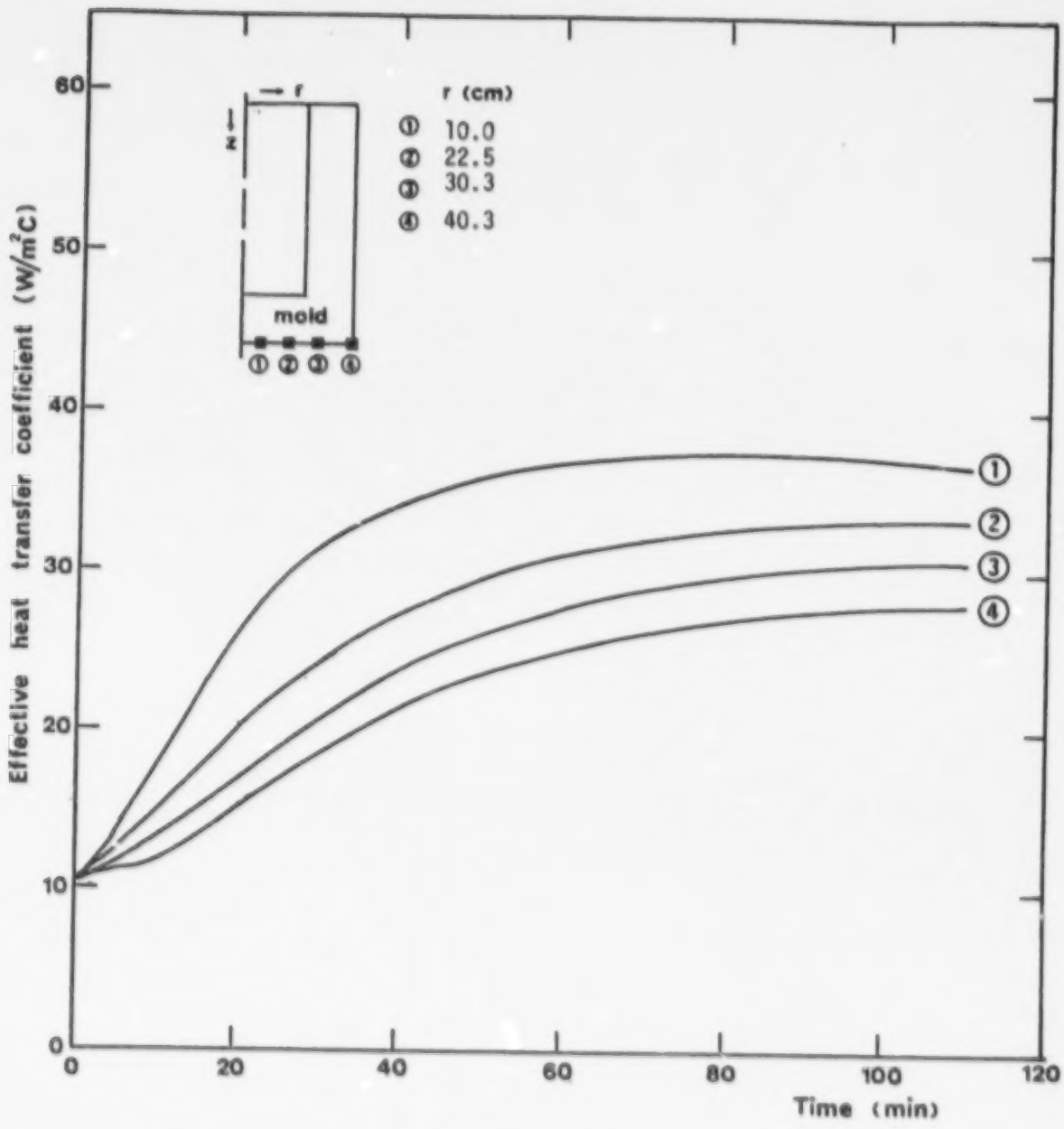


Fig. 4-12 Variation of effective heat transfer coefficient at bottom outer surface of mold.

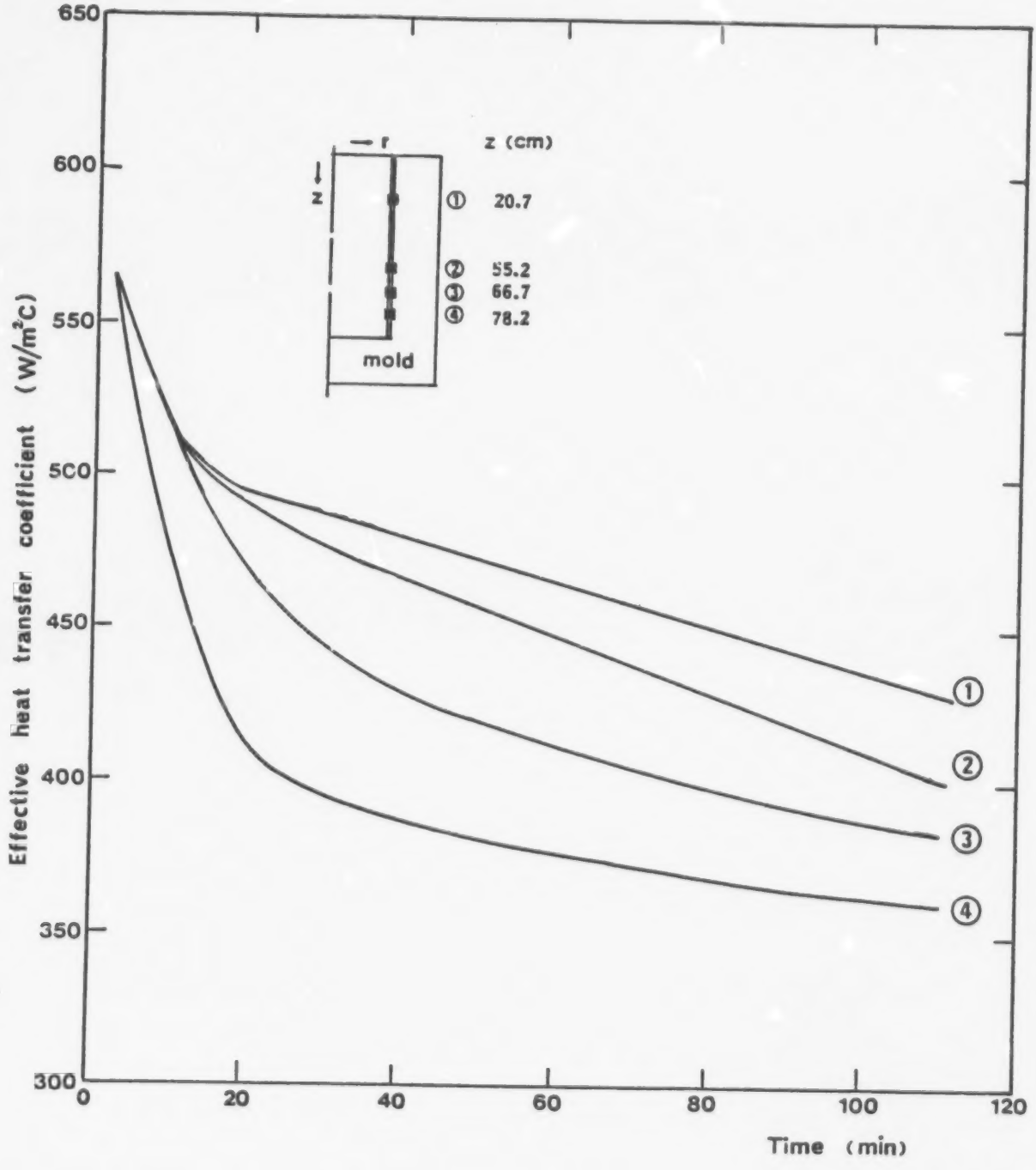


Fig. 4-13 Variation of the interface heat transfer coefficient.

CHAPTER 5

CONCLUSIONS

A mathematical model has been developed to investigate the solidification process in a casting. A finite-difference scheme using the alternating direction implicit method, was used to solve the heat transfer equations. A computer program has been written for a simple two-dimensional geometry defined in both cartesian and cylindrical polar coordinate systems. The program was tested for a simple geometry and gave reasonable results.

Further work will include the modelling of the complex shape mold, being presently considered at EMR, and the development of a three-dimensional program.

BIBLIOGRAPHY

1. ADAMS, C.M., Proc. Seminar, Liquid metals and solidification, 39th National Metal Congress (American Society of Metals, Chicago), 1957.

2. BALLANTYNE, A.S. and MITCHELL, A., "Modelling of Ingot Thermal Field in Consumable Electrode Remelting Processes", Ironmaking and Steelmaking, No. 4, p. 222, 1977.

3. BRODY, H.D. and STOEHR, R.A., "Computer Simulation of Heat Flow in Casting", J. of Metals, p. 20, Sept. 1980.

4. De FLEURY, R., Transaction AFS, (33), p. 235, 1925.

5. DEN HARTOG, H.W., RABENGERG, J.M. and WILLEMSE, J., "Application of a Mathematical Model in the Study of the Ingot Solidification Process", Ironmaking and Steelmaking, No. 2, p. 134, 1975.

6. DUSINBERRE, G.M., Heat Transfer Calculations by Finite Differences, Int. Text Book Co., 1961.

7. FAHRQUAR, R.B., Transaction AFS, (29), p. 171, 1920.

8. FLEMINGS, M.C., Solidification Processing, McGraw-Hill, 1974.

9. GARCIA, A. and PRATES, M., "Mathematical Model for the Unidirectional Solidification of Metals: I. Cooled Molds", Met. Trans. B., Vol. 9B, p. 449, Sept. 1978.

10. GARCIA, A. and PRATES, M., "Mathematical Model for the Unidirectional Solidification of Metals: II Massive Molds", Met. Trans. B., Vol. 10B, p. 85, March 1979.

11. HEUSER, J., "Finite Element Method for Thermal Analysis", NASA TND-7274, Nov. 1973.

12. KUSHNER, A.S. and WALSTON, W.H. Jr., "Conduction and Natural Convection Heat Transfer in a Phase Change Region", Computational Techniques for Interface Problems, presented at the Winter Annual Meeting of ASME, San Francisco California, Dec. 10-15, 1978.

13. LAZARIDIS, A., "A Numerical Solution of the Multidimensional Solidification (or Melting) Problem", Int. J. Heat Mass Transfer, Vol. 13, p. 1459, 1970.

14. LAZARIDIS, DAY and ZIMMERMANN, "A Computer-Aided Solution of the Solidification Problem", AFS Cast Metal Research Journal, p. 48, March 1974.

15. LIGHTFOOT, N.M.H., J. Iron Steel Inst., Vol. 119, p. 364, 1929.

16. LIGHTFOOT, N.M.H., J. Iron Steel Inst., Vol. 2, p. 162, 1932.

17. MARRONE, R.E., WILKES, J.O., and PEHLKE, R.D., "Numerical Simulation of Solidification, Part I: Low Carbon Steel Casting - 'T' Shape", Cast Metals Research Journal, p. 184, 1970.

18. MARRONE, R.E., WILKES, J.O., and PEHLKE, R.D., "Numerical Simulation of Solidification, Part II: Low Carbon Steel Casting - 'L' Shape", Cast Metals Research Journal, p. 188, 1970.

19. MORGAN, K., "A Numerical Analysis of Freezing and Melting with Convection", Computer Methods in Applied Mechanics and Engineering, 28, p. 275, 1981.

20. NISHIDA, Y., IMAI, T. and MATSUBARA, H., "Numerical Analysis of Solidification of Cast Iron in Metal Moulds", Trans. Iron Steel Inst. Japan, No. 13, p. 304, 1977.

21. PASCHKIS, V., "Studies on Solidification of Castings - Heat Transfer Committee Report", Trans. American Foundrymen's Assoc., Vol. 53, p. 90-101, 1945.

22. PASCHKIS, V., "Studies on the Solidification of Aluminum Castings", p. 366-371; "Studies of Solidification of White Iron Castings", p. 371-373, Trans. American Foundrymen's Society, Vol. 56, 1948.

23. PASCHKIS, V. and RYDAR, F.L., Direct Analog Computers, John Wiley & Sons, New York, 1968.

24. PEACEMAN, D.W., and RACHFORD, H.H., "The Numerical Solution of Parabolic and Elliptic Differential Equations", J. Soc. Indust. Appl. Math., Vol. 3, p. 28, 1955.

25. PEEL, D.A. and PENGELLY, A.E. Mathematical Models in Metallurgical Process Development, p. 186, The Iron and Steel Institute, London, 1970.

26. PEHLKE, R.D., KIRT, M.J., MARRONE, R.E., and COOK, D.J. "Numerical Simulation of Casting Solidification", Cast Metals Research Journal, Vol. 9, No. 2, p. 49, 1973.

27. RAYCHAUDHURI, B.C., CHATTERJEE, G., YADAV, K.G., and MAJUMDER, J., "Thermal Aspect of Steel Ingot Solidification and Numerical Investigation of a Two-dimensional Heat Flow Problem", Indian J. of Technology, Vol. 15, p. 185, 1977.

28. SALCUDEAN, M., LOW, C.H., "Mathematical Modelling of Heat Transfer in Permanent Moulds", University of Ottawa, 1982.

29. SARJANT, R.J. and SLACK, M.R., "Internal Temperature Distribution in the Cooling and Reheating of Steel Ingots", J. of Iron and Steel Inst., Vol. 177, p. 428, 1954.

30. SMITH, G.D., "Numerical Solution of Partial Differential Equations: Finite Difference Methods", Oxford University press, 1978.

31. SPENCER, S., CARLESS, P. and MAGEE, E., "Mathematical Model for Simulation of Solidification and Cooling of Cast Rolls", Ironmaking and Steelmaking, No. 3, 1981.

32. SRINIVASAN, M.N., "Analytical Studies on the Solidification of Hyperectectic Cast Iron in Metallic Molds", AFS Cast Metals Research Journal, p. 91, Sept. 1975.

33. SZEKELY, J. and JASSAL, A.S., "An Experimental and Analytical Study of the Solidification of a Binary Dendritic System", Met. Trans B. Vol. 9B, p. 389, 1978.

34. THAMBAN, M.I. and PANCHANATHAN, V., "Numerical Simulation of Solidification of Aluminum Alloys in Cast Iron Molds", AFS Transactions, p. 167, 1980.

35. WECKMAN, D.C., PICK. R.J., and NIESSEN, P., "A Numerical Heat Transfer Model of the D.C. Continuous Casting Process", Z. Metallkele, 70, p. 750, 1979.

36. WILSON, E. and NICKELL, R.E., "Application of the Finite Element Method to Heat Conduction Analysis", Nuclear Eng. and Design, Vol. 4, No. 3, p. 276, 1966.

APPENDIX

Listing of computer program for testing two-dimensional solidification in a cylindrical polar and cartesian coordinate system.

```
...  LEVEL  21          MAIN          DATE = 83179          13/34/53
C
C     ------------------------------------------------------------
C     |                                                          |
C     |  COPYRIGHT OF THIS PROGRAM IS THE PROPERTY OF:           |
C     |                                                          |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING, |
C     |                                                          |
C     |  UNIVERSITY OF OTTAWA.                                   |
C     |                                                          |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO|
C     |                                                          |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:         |
C     |                                                          |
C     |  DR. M. SALCUDEAN.                                       |
C     ------------------------------------------------------------
C
C     2D CYLINDRICAL - CALL LINE1, LINE2
C     SOLIDIFIED IN ABOUT 72 MIN. OR 4320.0 SEC
C
C     THIS PROGRAM CONSISTS OF MAIN PROGRAM AND 17 SUBPROGRAMS.
C
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STEAN,RCLIQ,
     * RCSOL,RCMOLD,RFUS,HF(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
C     * M A I N  P R O G R A M *
C     -------------------------
C
      CALL INIT1
      SCX=18.0/XAD(IBAR)
      SCZ=SCX
      CALL PLOTS(ZAD(KBAR)*3CZ+1.0)+1.0,17.5)
      CALL UPDATE
      CALL NCOAL
      CALL HTOP(1,IBAR)
3.0   CALL LINE1
      IF(ICYCLE.LE.1) CALL PRINT(3)
      CALL LINE2
      IF(ICYCLE.LE.1) CALL PRINT(4)
      CALL UPDATE
      ICYCLE = ICYCLE+1
      TIME = TIME+2.*DT
      IF(TIME-TPRINT) 2.0,2.1,2.1
2.1   CALL PRINT(2)
      IF(TIME.GE.3600.0) GO TO 2.2
      TPRINT=TPRINT+240.0
      GO TO 2.0
2.2   IF(TIME.GE.6.0E0) GO TO 2.3
      TPRINT=TPRINT+300.0
      GO TO 2.0
2.3   TPRINT=TPRINT+600.0
2.0   CO
```

```
G Level  21                    MAIN                    DATE = 83178          14/34/53

      IF(TIME-TPLOT) 302,303,303
  303 CALL TPLOT
      IF(TIME.GE.6[illegible]) GO TO 301
      TPLOT=TPLOT+5[illegible]
      GO TO 302
  301 TPLOT=TPLOT+5.E+6
  302 CONTINUE
      IF(TIME.GE.TSTOP) GO TO 6[illegible]
      GO TO 3[illegible]
 6[illegible] WRITE(6,6[illegible])
 6[illegible] FORMAT(/,15X,1[illegible]('*'),/,30X,' COMPUTATION IS ENDED WHEN TIME IS GR
     *EATER OR EQUAL TO  TSTOP',/,15X,1[illegible]('*'),//)
      CALL PLOT([illegible],[illegible],999)
      STOP
      END
```

```
V LEVEL  21                    MAIN                    DATE = 83276          19/34/53

      S U B P R O G R A M S .
      -----------------------

       SUBROUTINE INITT
      ------------------------------------------------------------------
      |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:               |
      |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,        |
      |  UNIVERSITY OF OTTAWA.                                          |
      |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO       |
      |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                |
      |  DR. M. SALCUDEAN.                                              |
      ------------------------------------------------------------------

       THIS SUBPROGRAM READS THE INITIAL VALUES FOR VARIABLES.
       THE INITIAL TEMPERATURE AT THE METAL-MOLD INTERFACE IS
       CALCULATED.
       SOME GEOMETRICAL CHARACTERS ARE ALSO COMPUTED.

       COMMON /GEOM/ DX(25),XAD(50),DZ(50),ZAD(50),FX(25,50),
      * FZ(25,50)

       COMMON /TEMP/ T(25,50),TN(25,50),TPOUR,TMELT,TAMB,TLIQ,TSOL,
      * TMOLD,TS(25,50)

       COMMON /PROP/ RHOS,STEAN,RELIQ,
      * RHOL,RHOMOLD,HFUS,HT(25),
      * HTC,HT

       COMMON /VAR/ IMAX,KMAX,IFACE,KFACE,IFN,IFP,KFN,KFP,IC(25,50),
      * ICOOL,ICYCLE,CT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL

       IMAX=25
       KMAX=50
       IFACE=14
       IFN=IFACE-1
       IFP= IFACE+1
       KFACE=4
       KFN=KFACE-1
       KFP= KFACE+1
       ICYCLE=0
       CYL=1.0
       TIME IN SECONDS
       CT=0.0
       TIME=0.0
       TPRINT=30.0
       TPLOT=0.0
       TSTOP=480.0
       TGAP=12.0

       PROPERTIES

       HEAT OF FUSION IN  J/KG
       HFUS= 209300.0
```

```
J LEVEL  21                    INITT                    DATE = 83179          13/34/53
C
C       TEMPERATURE IN  C
        TPCUR= 1383.0
        TAMB= 23.0
        TLIQ= 1498.88
        TSOL=1447.89
        TMOLD=121.0
C
C       EMISSIVITY
        EMIS= 0.8
C
C       STEFAN-BOLTZMANN CONSTANT IN  W/M**2.K**4
        STFAN= 5.67E-8
C
C       DENSITY IN   KG/M**3
        RHOLIQ=72...0
        RHOSOL=7467.6
        RHOMLD=7466.6
C
C       MESH SIZE IN  METERS
C
        READ(5,20) (DX(I),I=1,IBAR)
        READ(5,20) (DZ(K),K=1,KBAR)
 20     FORMAT(12F6.3)
        XAC(1)=1.0E-.7*DX(1)*1.0E-04
        ZAC(1)=0.0
        DO 22 I=2,IBAR
 22     XAC(I)= XAC(I-1)+(DX(I-1)+DX(I))/2.
        DO 23 K=2,KBAR
 23     ZAC(K)= ZAC(K-1)+(DZ(K-1)+DZ(K))/2.
        WRITE(6,30) (I,DX(I),I,XAC(I),I=1,IBAR)
        WRITE(6,31) (K,DZ(K),K,ZAC(K),K=1,KBAR)
 30     FORMAT(///,12(5X,'DX(',I2,')= ',F1..3,2X,'M',3X,
     * 'XAC(',I2,')= ',F1..3,2X,'M'/))
 31     FORMAT(///,12(5X,'DZ(',I2,')= ',F1..3,2X,'M',3X,
     * 'ZAC(',I2,')= ',F1..3,2X,'M'/))
C
        IBARM=IBAR-1
        DO 40 I=1,IBARM
 40     FX(I,I+1)=DX(I+1)/(DX(I)+DX(I+1))
        KBARM=KBAR-1
        DO 41 K= 1,KBARM
 41     FZ(K,K+1)=DZ(K+1)/(DZ(K)+DZ(K+1))
C
        DO 120 K=1,KFACE
        DO 130 I=1,IFACE
 130    T(I,K)=TPCUR
        DO 140 I=IFP,IBAR
 140    T(I,K)=TMOLD
 120    CONTINUE
        DO 150 K=KFP,KBAR
        DO 150 I=1,IBAR
 150    T(I,K)=TMOLD
        IF(TIME.GT.0.0) GO TO 160
        DO 200 K=1,KFACE
        T1= TPCUR-TMOLD
        T2= 1.0+COND(IFACE,K,3,1)/COND(IFACE,K,2,1)*
     *  (((COND(IFACE,K,2,1)*RO(IFACE,K,3,1)*SH(IFACE,K,3,1))/
     *  (COND(IFACE,K,3,1)*RO(IFACE,K,2,1)*SH(IFACE,K,2,1)))**0.5)
 200    T(IFACE,K)= TMOLD+T1/T2
        DO 210 I=1,IFACE
        T1= TPCUR-TMOLD
```

```
1 (3 Level  22                    SHIFT                 CATS = 83179          16/34/53

      T4= 1.4*CONC(I,KFACE,3,1)/CONC(I,KFACE,2,1)*
     * (((CLND(I,KFACE,2,1)*RC(I,KFACE,3,1)*SH(I,KFACE,3,1))/
     * (CONC(I,KFACE,3,1)*RC(I,KFACE,2,1)*SH(I,KFACE,2,1)))**0.5)
   21.  T(I,KFACE)= TNCLD*T1/T4
   18.  CONTINUE
C
      RETURN
      END
```

```
C 3 LEVEL  21                    MAIN                    DATE = 83176          15/34/53
C
C     SUBROUTINE NODAL
C    ------------------------------------------------------------------
C    |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:               |
C    |                                                                 |
C    |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,        |
C    |                                                                 |
C    |  UNIVERSITY OF OTTAWA.                                          |
C    |                                                                 |
C    |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO       |
C    |                                                                 |
C    |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                |
C    |                                                                 |
C    |  DR. M. SALCUDEAN.                                              |
C    ------------------------------------------------------------------
C
C     THIS SUBPROGRAM IDENTIFIES THE TYPE OF NODES FOR EACH
C     ONE OF THE GRID POINTS.
C
      COMMON /GEOM/ DX(2),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(2,3),TN(25,50),TFOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,3)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * RCSOL,RCMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFP,IFP,KFP,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TJ,TJP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DO 1000 K=1,KBAR
      DO 1000 I=1,IBAR
      IF(I.EQ.1) IC(I,K)=7
      IF(I.EQ.1.AND.K.EQ.1) IC(I,K)=8
      IF(I.EQ.1.AND.K.EQ.KFACE) IC(I,K)=4
      IF(I.EQ.1.AND.K.EQ.KBAR) IC(I,K)=1.
C
      IF(I.GT.1.AND.I.LT.IFP) IC(I,K)=2
      IF((I.GT.1.AND.I.LT.IFP).AND.K.EQ.1) IC(I,K)=5
      IF((I.GT.1.AND.I.LT.IFP).AND.K.EQ.KFACE) IC(I,K)=12
      IF((I.GT.1.AND.I.LT.IFP).AND.K.EQ.KBAR) IC(I,K)=3
C
      IF(I.EQ.IFP) IC(I,K)=13
      IF(I.EQ.IFP.AND.K.EQ.1) IC(I,K)=17
      IF(I.EQ.IFP.AND.K.EQ.KFACE) IC(I,K)=16
      IF(I.EQ.IFP.AND.(K.GT.KFACE.AND.K.LT.KBAR)) IC(I,K)=1
      IF(I.EQ.IFP.AND.K.EQ.KBAR) IC(I,K)=3
C
      IF(I.EQ.IFACE) IC(I,K)=11
      IF(I.EQ.IFACE.AND.K.EQ.1) IC(I,K)=14
      IF(I.EQ.IFACE.AND.K.EQ.KFACE) IC(I,K)=13
      IF(I.EQ.IFACE.AND.(K.GT.KFACE.AND.K.LT.KBAR)) IC(I,K)=1
      IF(I.EQ.IFACE.AND.K.EQ.KBAR) IC(I,K)=3
C
      IF(I.GT.IFACE.AND.I.LT.IBAR) IC(I,K)=1
      IF((I.GT.IFACE.AND.I.LT.IBAR).AND.K.EQ.1) IC(I,K)=5
      IF((I.GT.IFACE.AND.I.LT.IBAR).AND.K.EQ.KBAR) IC(I,K)=3
```

```
[illegible] LEVEL  21                    VOCAL                    DATE = 83179          13/34/53
      C
            IF(I.EQ.IBAR) IC(I,N)=2
            IF(I.EC.ISAR.AND.P.EQ.1) IC(I,P)=5
            IF(I.EQ.IBAR.AND.M.E).NEAR) IC(I,P)=4
      C
       [illegible] CONTINUE
            RETURN
            END
```

```
C LEVEL  21                 MAIN                 DATE = 83179          13/34/53
C
C
      SUBROUTINE UPDATE
C-----------------------------------------------------------------------
C  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:
C
C  DR. V. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,
C
C  UNIVERSITY OF OTTAWA.
C
C  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO
C
C  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:
C
C  DR. V. SALCUDEAN.
C-----------------------------------------------------------------------
C
C     THIS SUBPROGRAM REPLACES PRECEDING TEMPERATURE FIELD
C     BY THE MOST RECENT ONE OBTAINED FROM CALCULATIONS.
C
      COMMON /GEOM/ DA(25),XAC(25),DZ(3,1),ZAC(50),FA(25,50),
     * FZ(25,3,1)
C
      COMMON /TEMP/ T(27,3,1),TN(25,3,1),TMCUR,TMELT,TAMB,TLIQ,TSOL,
     - TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * ROSOL,ROMOLD,PFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFP,IFP,KFP,KFP,IC(25,50),
     * ICODE,ICYCLE,CT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DO 100 K=1,KBAR
      DO 100 I=1,IBAR
      TN(I,K)=T(I,K)
      RETURN
      END
```

```
LEVEL  21                    MAIN                    DATE = 83179          13/34/53
C
      FUNCTION CX(I,K,M,L)
C     ------------------------------------------------------------------
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:               |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,        |
C     |  UNIVERSITY OF OTTAWA.                                          |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO       |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                |
C     |  DR. M. SALCUDEAN.                                              |
C     ------------------------------------------------------------------
C
C     THIS SUBPROGRAM AS WELL AS THE ONE RIGHT AFTER, EMPLOYED
C     TO CALCULATE THE VALUES OF INTERMEDIATE VARIABLE;
C     CX = CONDUCTIVITY * 2./(DX(I)+DX(I+1)).
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TFCUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * ROSOL,ROMOLD,HFUS,HF(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,FFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
C
      GO TO (1.,2.,3.),L
1.    CX=2./((1.-FX(I,I+1))/COND(I,K,M,L)+FX(I,I+1)/
     *       COND(I+1,K,M,L))/(DX(I)+DX(I+1))
      RETURN
3.    CX=2./((1.-FX(I,I+1))/COND(I,K,3,L)+FX(I,I+1)/
     *       COND(I+1,K,3,L))/(DX(I)+DX(I+1))
      RETURN
      END
```

```
3 LEVEL  21                    MAIN                    DATE = 83178          18/34/53
C
C
      FUNCTION CZ(I,K,J,L)
C
C     ------------------------------------------------------------------
C     |                                                                |
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
C     |                                                                |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
C     |                                                                |
C     |  UNIVERSITY OF OTTAWA.                                         |
C     |                                                                |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
C     |                                                                |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
C     |                                                                |
C     |  DR. M. SALCUDEAN.                                             |
C     |                                                                |
C     ------------------------------------------------------------------
C
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * ROSOL,ROMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,LFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      GO TO (1..,2..,3..,4..),K
C
 1..  CZ= 2./((1.-FZ(K,K+1))/COND(I,K,J,L)+FZ(K,K+1)/
     *       COND(I,K+1,J,L))/(DZ(K)+DZ(K+1))
      RETURN
 3..  CZ= 2./((1.-FZ(K,K+1))/COND(I,K,J,L)+FZ(K,K+1)/
     *       COND(I,K+1,J,L))/(DZ(K)+DZ(K+1))
      RETURN
      END
```

```
0 LEVEL  2A                MAIN              DATE = 83179          19/34/53
C
C
      FUNCTION HS(K,L)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:                           C
C                                                                             C
C  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,                    C
C  UNIVERSITY OF OTTAWA.                                                      C
C                                                                             C
C  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO                   C
C  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                            C
C                                                                             C
C  DR. M. SALCUDEAN.                                                          C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     COMPUTE THE SIDE WALL HEAT TRANSFER COEFFICIENT USING
C     TEMPERATURES FROM PREVIOUS COMPUTATIONS.
C
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPCUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RGLIQ,
     * RGSOL,RGMOLD,HFUS,HT(25),
     * HTC,HTW
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
C
      GO TO (1,2,3),L
    1 TE= T(IBAR,K)
      GO TO 500
    2 TE= TN(IBAR,K)
      GO TO 500
    3 TE= TS(IBAR,K)
  500 CONTINUE
      SLEN= ZAD(KBAR)
      DELT= TE-TAMB
      HS1= 1.42*(DELT/SLEN)**0.25
      HS= HS1+EMIS*STFAN*((TE+273.15)**2+(TAMB+273.15)**2
     *     )*((TE+273.15)+(TAMB+273.15))
      RETURN
      END
```

```
3 LEVEL  21                    HAIN                DATE = 83178          18/34/33
      C
      C
            FUNCTION HB(I,L)
      C     ------------------------------------------------------------------
      C     |                                                                |
      C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
      C     |                                                                |
      C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
      C     |                                                                |
      C     |  UNIVERSITY OF OTTAWA.                                         |
      C     |                                                                |
      C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
      C     |                                                                |
      C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
      C     |                                                                |
      C     |  DR. M. SALCUDEAN.                                             |
      C     ------------------------------------------------------------------
      C
      C           COMPUTE THE BOTTOM WALL HEAT TRANSFER COEFFICIENT USING
      C           TEMPERATURES FROM PREVIOUS COMPUTATIONS.
      C
            COMMON /GRID/ DX(25),XAD(25),DZ(30),ZAD(30),FX(25,30),
           * FZ(25,30)
      C
            COMMON /TEMP/ T(25,30),TN(25,30),TPCUR,TMELT,TAMB,TLIQ,TSOL,
           - TMOLD,TS(25,30)
      C
            COMMON /PROP/ EMIS,STFAN,ROLIQ,
           * RQSOL,RQMOLD,HFUS,HF(25),
           * HTC,HTM
      C
            COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFN,IFP,KFN,KFP,IC(25,30),
           * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
      C
      C
            GO TO (1,2,3),L
          1 TE= T(I,KBAR)
            GO TO 50
          2 TE= TN(I,KBAR)
            GO TO 50
          3 TE= TS(I,KBAR)
         50 CONTINUE
            BLEN= XAD(IBAR)
            DELT= TE-TAMB
            ALT=DELT/BLEN/BLEN
            IF(ALT.LE.0.0) WRITE(6,99) I,TE,DELT,BLEN,ALT
         99 FORMAT(/,' ERROR IN HB ',2I4,4G12.5)
            HB1= 1.31*(DELT/BLEN/BLEN)**0.2
            HB= HB1+EMIS*STFAN*((TE+273.15)**2+(TAMB+273.15)**2
           *       )*((TE+273.15)+(TAMB+273.15))
            RETURN
            END
```

```
J LEVEL  22                    MAIN                    DATE = 83176          13/34/53
C
      SUBROUTINE HTOP(I1,I2)
C     ------------------------------------------------------------------
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:               |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,        |
C     |  UNIVERSITY OF OTTAWA.                                          |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO       |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                |
C     |  DR. M. SALCUDEAN.                                              |
C     ------------------------------------------------------------------
C
C     COMPUTE THE TOP WALL HEAT TRANSFER COEFFICIENT USING
C     TEMPERATURES FROM PREVIOUS COMPUTATIONS.
C     IF TOP BOUNDARY IS INSULATED, GO TO 2000
C
      COMMON /GECM/ DX(25),XAD(25),DZ(50),ZAC(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TFOUR,TMELT,TAMB,TLIQ,TSOL,
     * TNOLD,TS(25,5.)
C
      COMMON /PROP/ EMIS,STFAN,ROLIQ,
     * RCSOL,RCPCLD,HFUS,HT(25),
     * HTC,HTN
C
      COMMON /VAR/ IBAR,KBAR,IFACE,NFACE,IFA,IFB,KFA,KFB,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DO 2000 I=I1,I2
 2000 HT(I)= 0.0
      HTC= 0.0
      HTN= 0.0
      RETURN
  777 IF(TIME.LT.TGAP) GO TO 30.
      TLEN1= XAD(IFA)
      DO 100 I=1,IFB
      DELT= TN(I,1)-TAMB
      HT1= 1.32*(DELT/TLEN1)**0.25
      HT(I)= HT1+EMIS*STFAN*((TN(I,1)+273.15)**2+(TAMB+273.15)**2
     *      )*((TN(I,1)+273.15)+(TAMB+273.15))
  100 CONTINUE
      TLEN2= XAD(IBAR)-XAD(IFACE)
      DO 200 I=IFACE,IBAR
      DELT= TN(I,1)-TAMB
      HT2= 1.32*(DELT/TLEN2)**0.25
      HT(I)= HT2+EMIS*STFAN*((TN(I,1)+273.15)**2+(TAMB+273.15)**2
     *      )*((TN(I,1)+273.15)+(TAMB+273.15))
  200 CONTINUE
C2000 CONTINUE
      RETURN
   30. CONTINUE
      TLEN3= XAD(IFACE)
      DO 301 I=1,IFB
      DELT= TN(I,1)-TAMB
```

```
IV 3 LEVEL  2a                    HTOP                    DATE = 83179          13/34/53

      HT3= 1.32*(DELT/TLEN3)**.25
      HT(I)= HT3*EMIS*STFAN*((TN(I,1)+273.15)**2+(TAMB+273.15)**2
     *     )*((TN(I,1)+273.15)+(TAMB+273.15))
  302 CONTINUE
      TLEN4= RAD(IBAR)-RAD(IFACE)
      DO 303 I=IFACE,IBAR
      DELT= TN(I,1)-TAMB
      HT4= 1.32*(DELT/TLEN4)**.25
      HT(I)= HT4*EMIS*STFAN*((TN(I,1)+273.15)**2+(TAMB+273.15)**2
     *     )*((TN(I,1)+273.15)+(TAMB+273.15))
  303 CONTINUE
      TLENC= RAD(IFACE)
      TLENM= RAD(IBAR)-RAD(IFACE)
      DELT= TN(IFACE,1)-TAMB
      HTC1= 1.32*(DELT/TLENC)**.25
      HTC= HTC1*EMIS*STFAN*((TN(IFACE,1)+273.15)**2+(TAMB+273.15)
     *     **2)*((TN(IFACE,1)+273.15)+(TAMB+273.15))
      HTM1= 1.32*(DELT/TLENM)**.25
      HTM= HTM1*EMIS*STFAN*((TN(IFACE,1)+273.15)**2+(TAMB+273.15)
     *     **2)*((TN(IFACE,1)+273.15)+(TAMB+273.15))
 3350 CONTINUE
      RETURN
      END
```

```
1 S LEVEL  21                MAIN              DATE = 83178          18/34/53
C
C
      FUNCTION HG(K,L)
C     ------------------------------------------------------------------
C     |                                                                |
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
C     |                                                                |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
C     |                                                                |
C     |  UNIVERSITY OF OTTAWA.                                         |
C     |                                                                |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
C     |                                                                |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
C     |                                                                |
C     |  DR. M. SALCUDEAN.                                             |
C     |                                                                |
C     ------------------------------------------------------------------
C
C     COMPUTE THE GAP HEAT TRANSFER COEFFICIENT USING
C     TEMPERATURES FROM PREVIOUS COMPUTATIONS.
C
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAC(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPOUR,TMELT,TAMB,TLIQ,TSOL,
     * TNOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * ROSOL,RCPOLD,HFUS,HT(25),
     * HTC,HTN
C
      COMMON /VAR/ IBAN,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,LT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      GO TO (1,2,3),L
    1 TE1= T(IFM,K)
      TE2= T(IFACE,K)
      GO TO 500
    2 TE1= TN(IFM,K)
      TE2= TN(IFACE,K)
      GO TO 500
    3 TE1= TS(IFM,K)
      TE2= TS(IFACE,K)
  500 CONTINUE
      DGAP= (DX(IFM)+DX(IFACE))/2.
      HG1= 0.05*CCND(IFM,K,1,L)/DGAP
      HG= HG1+EMIS*STFAN*((TE1+273.15)**2+
     *          (TE2+273.15)**2)*((TE1+273.15)+
     *          (TE2+273.15))
      RETURN
      END
```

```
[ 3 LEVEL  21                    MAIN                    DATE = 83179          18/34/53
C
C
      SUBROUTINE LINEX
C     ------------------------------------------------------------------
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:               |
C     |                                                                 |
C     |  DR. P. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,        |
C     |                                                                 |
C     |  UNIVERSITY OF OTTAWA.                                          |
C     |                                                                 |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO       |
C     |                                                                 |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                |
C     |                                                                 |
C     |  DR. P. SALCUDEAN.                                              |
C     ------------------------------------------------------------------
C
C     COMPUTE THE NON-ZERO COEFFICIENTS OF THE SYSTEM OF LINEAR
C     EQUATIONS FOR LINE ITERATIONS.
C     THE SUBPROGRAM TRID IS CALLED TO SOLVE THE ABOVE EQUATIONS FOR
C     TEMPERATURE FIELD AT THE FIRST HALF OF TIME-STEP ELAPES.
C
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPCUR,TMELT,TAMB,TLIQ,TSOL,
     * THOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RHCLIQ,
     * RHOSOL,RHOCLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DIMENSION AA(50),DD(50),CC(50),BB(50)
C
      L=2
      DO 5500 K=1,KBAR
      DO 3000 I=1,IBAR
      IP=I+1
      IM=I-1
      KP=K+1
      KM=K-1
      IF(TN(I,K).LE.TSOL) M=1
      IF(TN(I,K).GE.TLIQ) M=2
      IF(TN(I,K).GT.TSOL.AND.TN(I,K).LT.TLIQ) M=4
      IF(I.GT.IFACE) M=3
      IF(K.GT.KFACE) M=3
      ICODE=IC(I,K)
C
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17), ICODE
C
    1 AA(I)= -(CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
      DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CX(IM,K,M,L)
     *       +CX(I,K,M,L))/DX(I)
     *        +CYL*(CX(I,K,M,L)-CX(IM,K,M,L))/(2.*XAD(I))
      CC(I)= -(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAD(I)))
      BB(I)=(RO(I,K,M,L)*SH(I,K,M,L)/DT-(CZ(I,KM,M,L)
     *       +CZ(I,K,M,L))/DZ(K))*TN(I,K)
```

```
[...]3 LEVEL  21                    LINEA                    DATE = 83179          19/34/53
     *          +(CZ(I,KM,M,L)/DZ(K))*TN(I,KM)+(CZ(I,K,M,L)/DZ(K))
     *          *TN(I,KP)
           GO TO 5000
C
    2      AA(I)= -(2.*CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/XAD(I))
           CC(I)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(CX(IM,K,3,L)
     *          +HS(K,L))/DX(I)-CYL*CX(IM,K,3,L)/XAD(I)
           DD(I)= 0.0
           BB(I)= CZ(I,KM,3,L)/DZ(K)*TN(I,KM)+CZ(I,K,3,L)/DZ(K)
     *          *TN(I,KP)+(RO(I,K,3,L)
     *          *SH(I,K,3,L)/DT-(CZ(I,KM,3,L)+CZ(I,K,3,L))
     *          /DZ(K))*TN(I,K)
     *          +2.*HS(K,L)/DX(I)*TAMB
           GO TO 5000
C
    3      AA(I)= -(CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/(2.*XAD(I)))
           DD(I)= RO(I,K,3,L)*SH(I,K,3,L)/DT+(CX(IM,K,3,L)
     *          +CX(I,K,3,L))/DX(I)
     *            +CYL*(CX(I,K,3,L)-CX(IM,K,3,L))/(2.*XAD(I))
           CC(I)= -(CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(2.*XAD(I)))
           BB(I)= 2.*CZ(I,KM,3,L)/DZ(K)*TN(I,KM)+2.*HB(I,L)/DZ(K)
     *          *TAMB+(RO(I,K,3,L)*SH(I,K,3,L)/DT-2.*(CZ(I,KM,3,L)
     *          +HB(I,L))/DZ(K))*TN(I,K)
           GO TO 5000
C
    4      AA(I)= -(2.*CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/XAD(I))
           DD(I)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(CX(IM,K,3,L)
     *          +HS(K,L))/DX(I)-CYL*CX(IM,K,3,L)/XAD(I)
           CC(I)= 0.0
           BB(I)= 2.*CZ(I,KM,3,L)/DZ(K)*TN(I,KM)+(RO(I,K,3,L)
     *          *SH(I,K,3,L)/DT-2.*(CZ(I,KM,3,L)+HB(I,L))/DZ(K))*TN(I,K)
     *          +2.*HS(K,L)/DX(I)*TAMB
     *          +2.*HB(I,L)/DZ(K)*TAMB
           GO TO 5000
C
    5      AA(I)= -(CX(IM,K,M,L)/DX(I)-CYL*CX(IM,M,M,L)/(2.*XAD(I)))
           DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CX(IM,K,M,L)
     *          +CX(I,K,M,L))/DX(I)
     *            +CYL*(CX(I,K,M,L)-CX(IM,K,M,L))/(2.*XAD(I))
           CC(I)= -(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAD(I)))
           BB(I)= (RO(I,K,M,L)*SH(I,K,M,L)/DT-2.*(HT(I)
     *            +CZ(I,K,M,L))/CZ(K))*TN(I,K)
     *        +2.*CZ(I,KM,M,L)/DZ(K)*TN(I,KP)+2.*HT(I)/DZ(K)*TAMB
           GO TO 5000
C
    6      AA(I)= -(2.*CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/XAD(I))
           DD(I)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(CX(IM,K,3,L)
     *          +HS(K,L))/DX(I)-CYL*CX(IM,K,3,L)/XAD(I)
           CC(I)= 0.0
           BB(I)= (RO(I,K,3,L)*SH(I,K,3,L)/DT-2.*(HT(I)
     *          +CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *       +2.*CZ(I,K,3,L)/DZ(K)*TN(I,KP)+2.*HT(I)/DZ(K)*TAMB
     *          +2.*HS(K,L)/DX(I)*TAMB
           GO TO 5000
C
    7      AA(I)= 0.0
           DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+2.*CX(I,K,M,L)/DX(I)
     *            +CYL*CX(I,K,M,L)/(1.*XAD(I))
           CC(I)= -(2.*CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(1.*XAD(I)))
           BB(I)= CZ(I,KM,M,L)/DZ(K)*TN(I,KM)+CZ(I,K,M,L)/DZ(K)
     *          *TN(I,KP)+(RO(I,K,M,L)*SH(I,K,M,L)/DT
     *          -(CZ(I,KM,M,L)+CZ(I,K,M,L))/DZ(K))*TN(I,K)
```

```
(V 3 LEVEL  21                    LINEX                    DATE = 83179          18/34/53

      GO TO 5000
C
   8  AA(I)= 0.0
      CD(I)=(RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))
     *     /DT+2.*(CX(I,K,M,L)+CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)+
     *          CX(I,K,3,L))/(1.*XAD(I))
      CC(I)=-(2.*(CX(I,K,M,L)+CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)+
     *          CX(I,K,3,L))/(1.*XAD(I)))
      BB(I)= 2.*CZ(I,KP,M,L)/DZ(K)*TN(I,KP)+
     *    ((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))/
     *   DT-2.*(CZ(I,KP,M,L)+CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *        +2.*CZ(I,K,3,L)/DZ(K)*TN(I,KP)
      GO TO 5000
C
   9  AA(I)= 0.0
      CD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+2.*CX(I,K,M,L)/DX(I)
     *          +CYL*CX(I,K,M,L)/(1.*XAD(I))
      CC(I)= -(2.*CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(1.*XAD(I)))
      BB(I)= (RO(I,K,M,L)*SH(I,K,M,L)/DT-2.*(CZ(I,K,M,L)
     *         +HT(I))/DZ(K))*TN(I,K)
     *    +2.*CZ(I,K,M,L)/DZ(K)*TN(I,KP)+2.*HT(I)/DZ(K)*TAMB
      GO TO 5000
C
  10  AA(I)= 0.0
      DD(I)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*CX(I,K,3,L)/DX(I)
     *          +CYL*CX(I,K,3,L)/(1.*XAD(I))
      CC(I)= -(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(1.*XAD(I)))
      BB(I)= 2.*CZ(I,KM,3,L)/DZ(K)*TN(I,KM)+(RO(I,K,3,L)
     *       *SH(I,K,3,L)/DT-2.*(CZ(I,KM,3,L)+HB(I,L))/DZ(K))*TN(I,K)
     *          +2.*HB(I,L)/DZ(K)*TAMB
      GO TO 5000
C
  11  IF(TIME.GE.TGAP) GO TO 511
      AA(I)= -(2.0*CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/XAD(I))
      DD(I)= (RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)
     *     *SH(I,K,3,L))/DT+2.0*(CX(IM,K,M,L)+CX(I,K,3,L))/DX(I)
     *          +CYL*(CX(I,K,3,L)-CX(IM,K,M,L))/XAD(I)
      CC(I)= -(2.0*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))
      BB(I)= (CZ(I,KM,M,L)+CZ(I,KM,3,L))/DZ(K)*TN(I,KM)
     *        +((RO(I,K,M,L)*SH(I,K,M,L)
     *         +RO(I,K,3,L)*SH(I,K,3,L))/DT-(CZ(I,KM,M,L)
     *          +CZ(I,KM,3,L)+CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *          +(CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K)*TN(I,KP)
      GO TO 5000
 511  AA(I)= -2.0*HG(K,L)/DX(I)
      DD(I)= RO(I,M,3,L)*SH(I,M,3,L)/DT+2.*(HG(K,L)
     *       +CX(I,K,3,L))/DX(I)+CYL*CX(I,K,3,L)/XAD(I)
      CC(I)= -(2.0*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))
      BB(I)= CZ(I,KM,3,L)/DZ(K)*TN(I,KM)+((RO(I,K,3,L)
     *     *SH(I,K,3,L))/DT-(CZ(I,KM,3,L)+CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *          +CZ(I,K,3,L)/DZ(K)*TN(I,KP)
      GO TO 5000
C
  12  AA(I)= -((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL
     *      *(CX(IM,K,M,L)+CX(IM,K,3,L))/(2.*XAD(I)))
      DD(I)=(RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))
     *       /DT+(CX(IM,K,M,L)+CX(IM,K,3,L)+CX(I,K,M,L)
     *       +CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)
     *        +CX(I,K,3,L)-CX(IM,K,M,L)-CX(IM,K,3,L))/
     *          (2.*XAD(I))
      CC(I)=-((CX(I,K,M,L)+CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)+
     *          CX(I,K,3,L))/(2.*XAD(I)))
```

```
3 LEVEL  21                    LINEA                    DATE = 83179          18/34/53
      BB(I)= 2.)*CZ(I,KM,M,L)/DZ(K)*TN(I,KM)+
     *  ((RC(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))/DT
     *          -2.*(CZ(I,KM,M,L)+CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *          +2.*CZ(I,K,3,L)/DZ(K)*TN(I,KP)
      GO TO 5000
C
  13  IF(TIME.GE.TGAP) GO TO 513
      AA(I)= -((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-
     *       CYL*(CX(IM,K,M,L)+CX(IM,K,3,L))/(2.*XAD(I)))
      DD(I)= (RO(I,K,M,L)*SH(I,K,M,L)+3.0*RC(I,K,3,L)
     *       *SH(I,K,3,L))/(2.*DT)
     *       +(CX(IM,K,M,L)+CX(IM,K,3,L)+2.*CX(I,K,3,L))/DX(I)
     *    +CYL*(2.*CX(I,K,3,L)-CX(IM,K,3,L)
     *     -CX(IM,K,3,L))/(2.*XAD(I))
      CC(I)= -(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))
      BB(I)= (CZ(I,KM,M,L)+CZ(I,KM,3,L))/DZ(K)*TN(I,KM)+
     *       2.*CZ(I,K,3,L)/DZ(K)*TN(I,KP)
     * +((RO(I,K,M,L)*SH(I,K,M,L)+3.*RC(I,K,3,L)*SH(I,K,3,L))
     * /(2.*DT)-(CZ(I,KM,M,L)+CZ(I,KM,3,L)+2.*CZ(I,K,3,L))/DZ(K)
     *       )*TN(I,K)
      GO TO 5000
 513  AA(I)=-((CX(IM,K,3,L)+HG(K,L))/DX(I)-CYL*CX(IM,K,3,L)
     *      /(2.*XAD(I)))
      DD(I)=(3.*RO(I,K,3,L)*SH(I,K,3,L))/(2.*DT)+(CX(IM,K,3,L)+
     *    HG(K,L)+2.*CX(I,K,3,L))/DX(I)+CYL*(2.*CX(I,K,3,L)-
     *       CX(IM,K,3,L))/(2.*XAD(I))
      CC(I)= -(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))
      BB(I)= CZ(I,KM,3,L)/DZ(K)*TN(I,KM)+((3.*RO(I,K,3,L)*
     *    SH(I,K,3,L))/(2.*DT)-(CZ(I,KM,3,L)+2.*CZ(I,K,3,L))/DZ(K))*
     *       TN(I,K)+2.*CZ(I,K,3,L)/DZ(K)*TN(I,KP)
      GO TO 5000
C
  14  IF(TIME.GE.TGAP) GO TO 514
      AA(I)= -(CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
      DD(I)= (RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))/
     *       (2.*DT)+(CX(IM,K,M,L)+CX(I,K,3,L))/DX(I)
     *      +CYL*(CX(I,K,3,L)-CX(IM,K,M,L))/(2.*XAD(I))
      CC(I)= -(CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(2.*XAD(I)))
      BB(I)=((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))/
     *    (2.*DT)-(HTC+HTM+CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *       +(CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K)*TN(I,KP)
     *       +(HTC+HTM)/DZ(K)*TAMB
      GO TO 5000
 514  AA(I)= -HG(K,L)/DX(I)
      DD(I)= RO(I,K,3,L)*SH(I,K,3,L)/(2.*DT)+(HG(K,L)
     *       +CX(I,K,3,L))/DX(I)+CYL*CX(I,K,3,L)/(2.*XAD(I))
      CC(I)= -(CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(2.*XAD(I)))
      BB(I)= (RO(I,K,3,L)*SH(I,K,3,L)/(2.*DT)-(HT(I)
     *       +CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *     +CZ(I,K,3,L)/DZ(K)*TN(I,KP)+HT(I)/DZ(K)*TAMB
      GO TO 5000
C
  15  IF(TIME.GE.TGAP) GO TO 515
      AA(I)= -(CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
      DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CX(IM,K,M,L)
     *       +CX(I,K,M,L))/DX(I)+CYL*(CX(I,K,M,L)-CX(IM,K,M,L))
     *       /(2.*XAD(I))
      CC(I)= -(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAD(I)))
      BB(I)=(RO(I,K,M,L)*SH(I,K,M,L)/DT-(CZ(I,KM,M,L)
     *       +CZ(I,K,M,L))/DZ(K))*TN(I,K)
     *       +(CZ(I,KM,M,L)/DZ(K))*TN(I,KM)+(CZ(I,K,M,L)/DZ(K))
     *       *TN(I,KP)
```

```
T. G LEVEL  21                    LINEX                DATE = 83179          18/34/53

      GO TO 5000
  515 AA(I)= -(2.*CX(IM,K,M,L)/DX(I)-CYL*CX(IP,K,M,L)/XAD(I))
      DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+2.*(CX(IM,K,M,L)
     *        +HG(K,L))/DX(I)-CYL*CX(IP,K,M,L)/XAD(I)
      CC(I)= -2.*HG(K,L)/DX(I)
      BB(I)= CZ(I,KP,M,L)/DZ(K)*TN(I,KM)+(RC(I,K,M,L)
     *       *SH(I,K,M,L)/DT-(CZ(I,KP,M,L)+CZ(I,K,M,L))/CZ(K))*TN(I,K)
     *         +CZ(I,K,M,L)/DZ(K)*TN(I,KP)
      GO TO 5000
C
   16 IF(TIME.GE.TGAP) GO TO 516
      AA(I)= -((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL
     *        *(CX(IM,K,M,L)+CX(IM,K,3,L))/(2.*XAD(I)))
      DD(I)=(RO(I,K,M,L)*SH(I,K,M,L)+RC(I,K,3,L)*SH(I,K,3,L))
     *      /DT+(CX(IM,K,M,L)+CX(IM,K,3,L)+CX(I,K,M,L)
     *      +CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)
     *      +CX(I,K,3,L)-CX(IP,K,M,L)-CX(IM,K,3,L))/(2.*XAD(I))
      CC(I)=-((CX(I,K,M,L)+CX(I,K,3,L))/CX(I)+CYL*(CX(I,K,M,L)+
     *         CX(I,K,3,L))/(2.*XAD(I)))
      BB(I)= 2.0*CZ(I,KM,M,L)/DZ(K)*TN(I,KM)+((RO(I,K,M,L)
     *       *SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))/DT-
     *         2.*(CZ(I,KM,M,L)+CZ(I,K,3,L))/DZ(K))*TN(I,K)
     *         +2.*CZ(I,K,3,L)/CZ(K)*TN(I,KP)
      GO TO 5000
  516 AA(I)=-((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL*(CX(IM,K,M,L)
     *         +CX(IM,K,3,L))/(2.*XAD(I)))
      DC(I)= (RO(I,K,M,L)*SH(I,K,M,L)+2.*RO(I,K,3,L)
     *       *SH(I,K,3,L))/(2.*DT)+
     *     (CX(IM,K,M,L)+CX(IM,K,3,L)+CX(I,K,3,L)+HG(K,L))/
     *      DX(I)+CYL*(CX(I,K,3,L)-CX(IM,K,M,L)-CX(IP,K,3,L))
     *      /(2.*XAD(I))
      CC(I)= -((CX(I,K,3,L)+HG(K,L))/DX(I)+CYL*CX(I,K,3,L)/
     *       (2.*XAD(I)))
      BB(I)= CZ(I,KM,M,L)/DZ(K)*TN(I,KM)+((RC(I,K,M,L)
     *      *SH(I,K,M,L)+2.*RO(I,K,3,L)*SH(I,K,3,L))/(2.*DT)
     *       -(CZ(I,KM,M,L)+2.*CZ(I,K,3,L))/CZ(K))*TN(I,K)
     *       +(2.0*CZ(I,K,3,L))/DZ(K)*TN(I,KP)
      GO TO 5000
C
   17 IF(TIME.GE.TGAP) GO TO 517
      AA(I)= -(CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
      DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CX(IM,K,M,L)
     *        +CX(I,K,M,L))/DX(I)
     *        +CYL*(CX(I,K,M,L)-CX(IM,K,M,L))/(2.*XAD(I))
      CC(I)= -(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAD(I)))
      BB(I)= (RO(I,K,M,L)*SH(I,K,M,L)/DT-2.*(HT(I)
     *       +CZ(I,K,M,L))/DZ(K))*TN(I,K)
     *       +2.*CZ(I,K,M,L)/DZ(K)*TN(I,KP)+2.*HT(I)/DZ(K)*TAMB
      GO TO 5000
  517 AA(I)= -(CX(IP,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
      DD(I)= RO(I,K,M,L)*SH(I,K,M,L)/(2.*DT)+(CX(IM,K,M,L)
     *       +HG(K,L))/DX(I)-CYL*CX(IP,K,M,L)/(2.*XAD(I))
      CC(I)= -HG(K,L)/DX(I)
      BB(I)= (RO(I,K,M,L)*SH(I,K,M,L)/(2.*DT)-(HT(I)
     *        +CZ(I,K,M,L))/DZ(K))*TN(I,K)
     *        +CZ(I,K,M,L)/DZ(K)
     *        *TN(I,KP)+HT(I)/DZ(K)*TAMB
C
 5000 CONTINUE
      CALL TRID(AA,DD,CC,BB,IBAR)
      DO 300 I=1,IBAR
  300 TS(I,K)=BB(I)
```

```
I  G LEVEL  21                    LINEX                    DATE = 83178          18/34/53

     5500  CONTINUE
           RETURN
           END
```

```
IV G LEVEL  21                    MAIN                    DATE = 83179          10/34/53
C
C
      SUBROUTINE LINEZ
C
C     ------------------------------------------------------------------
C     |                                                                |
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
C     |                                                                |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
C     |                                                                |
C     |  UNIVERSITY OF OTTAWA.                                         |
C     |                                                                |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
C     |                                                                |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
C     |                                                                |
C     |  DR. M. SALCUDEAN.                                             |
C     |                                                                |
C     ------------------------------------------------------------------
C
C
C     COMPUTE THE NON-ZERO COEFFICIENTS OF THE SYSTEM OF LINEAR
C     EQUATIONS FOR COLUMN ITERATIONS.
C
C     THE SUBROUTINE TRID IS CALLED TO SOLVE THE EQUATIONS FOR
C     TEMPERATURE FIELD AT THE SECOND HALF OF THE TIME-STEP ELAPES.
C
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPCUR,TMELT,TAME,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * RCSOL,RONCLC,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DIMENSION AA(60),DD(60),CC(60),BB(60)
      L=3
      DO 5500 I=1,IBAR
      DO 5500 K=1,KBAR
      IP=I+1
      IM=I-1
      KP=K+1
      KM=K-1
      IF(TS(I,K).LE.TSOL) M=1
      IF(TS(I,K).GE.TLIQ) M=2
      IF(TS(I,K).GT.TSOL.AND.TS(I,K).LT.TLIQ) M=4
      IF(I.GT.IFACE) M=3
      IF(K.GT.KFACE) M=3
      ICODE=IC(I,M)
C
C
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17), ICODE
C
    1 AA(K)= -CZ(I,KM,M,L)/DZ(K)
      DD(K)= RC(I,K,M,L)*SH(I,K,M,L)/DT+(CZ(I,KM,M,L)
     *       +CZ(I,K,M,L))/DZ(K)
      CC(K)= -CZ(I,K,M,L)/DZ(K)
      BB(K)= (CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
     *    *TS(IM,K)+(RC(I,K,M,L)*SH(I,K,M,L)/DT-(CX(IM,K,M,L)
```

```
V3 3 LEVEL  21                    LINEZ                    DATE = 83179          18/34/53
     *       +CX(I,M,K,L))/DX(I)-CYL*(CX(I,P,K,L)-CX(IM,K,M,L))/
     *     (2.*XAD(I)))
     *       *TS(I,K)+(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,P,L)/
     *       (2.*XAC(I)))*TS(IP,K)
      GO TO 5000
C
   2  AA(K)= -CZ(I,KM,3,L)/DZ(K)
      DD(K)= RO(I,K,3,L)*SH(I,K,3,L)/DT+(CZ(I,KM,3,L)
     *       +CZ(I,K,3,L))/DZ(K)
      CC(K)= -CZ(I,K,3,L)/DZ(K)
      BB(K)= (2.*CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/XAD(I))
     *     *TS(IP,K)+(RO(I,K,3,L)*SH(I,K,3,L)/DT-2.*
     *   (CX(IM,K,3,L)+HS(K,L))/DX(I)+CYL*CX(IM,K,3,L)/XAD(I)
     *    )*TS(I,K)
     *       +2.*HS(K,L)/DX(I)*TAMB
      GO TO 5000
C
   3  AA(K)= -2.*CZ(I,KM,3,L)/DZ(K)
      DD(K)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(CZ(I,KM,3,L)
     *       +HB(I,L))/DZ(K)
      CC(K)= 0.0
      BB(K)= (CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/(2.*XAD(I)))
     *     *TS(IP,K)+(RO(I,K,3,L)*SH(I,K,3,L)/DT-(CX(IM,K,3,L)
     *       +CX(I,K,3,L))/DX(I)
     *       -CYL*(CX(I,K,3,L)-CX(IM,K,3,L))/(2.*XAD(I)))*TS(I,K)
     *       +(CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(2.*XAC(I)))
     *       *TS(IP,K)
     *       +2.*HB(I,L)/DZ(K)*TAMB
      GO TO 5000
C
   4  AA(K)= -2.*CZ(I,KM,3,L)/DZ(K)
      DD(K)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(CZ(I,KM,3,L)
     *       +HB(I,L))/DZ(K)
      CC(K)= 0.0
      BB(K)= (2.*CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/XAD(I))
     *       *TS(IM,K)+(RO(I,K,3,L)*SH(I,K,3,L)/DT
     *       -2.*(CX(IM,K,3,L)+HS(K,L))/DX(I)+CYL*
     *       CX(IM,K,3,L)/XAD(I))*TS(I,K)
     *       +2.*HS(K,L)/DX(I)*TAMS
     *       +2.*HB(I,L)/DZ(K)*TAMB
      GO TO 5000
C
   5  AA(K)= 0.0
      DD(K)= RO(I,K,M,L)*SH(I,K,M,L)/DT+2.*(HT(I)
     *       +CZ(I,K,M,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,M,L)/DZ(K)
      BB(K)= (CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
     *     *TS(IM,K)+(RO(I,K,M,L)*SH(I,K,M,L)/DT-(CX(IM,K,M,L)
     *       +CX(I,K,M,L))/DX(I)
     *       -CYL*(CX(I,K,M,L)-CX(IM,K,M,L))/(2.*XAD(I)))*TS(I,K)
     *       +(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAC(I)))
     *       *TS(IP,K)+2.*HT(I)/DZ(K)*TAMB
      GO TO 5000
C
   6  AA(K)= 0.0
      DD(K)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(HT(I)
     *       +CZ(I,K,3,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,3,L)/DZ(K)
      BB(K)= (2.*CX(IM,K,3,L)/DX(I)-CYL*CX(IM,K,3,L)/XAD(I))
     *       *TS(IM,K)+(RO(I,K,3,L)*SH(I,K,3,L)/DT-2.*
     *   (CX(IM,K,3,L)+HS(K,L))/DX(I)+CYL*CX(IM,K,3,L)/XAD(I))*TS(I,K)
     *       +2.*HT(I)/DZ(K)*TAMB
```

```
I.  3 LEVEL  21                    LINEZ                 DATE = 83179          13/34/53
     *          +2.*HS(K,L)/DX(I)*TAMB
      GO TO 5000
C
   7  AA(K)= -CZ(I,KM,M,L)/DZ(K)
      DD(K)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CZ(I,KM,M,L)
     *          +CZ(I,K,M,L))/DZ(K)
      CC(K)= -CZ(I,K,M,L)/DZ(K)
      BB(K)= (RO(I,K,M,L)*SH(I,K,M,L)/DT-2.*CX(I,K,M,L)/DX(I)
     *          -CYL*CX(I,K,M,L)/(1.*XAD(I)))*TS(I,K)
     *         +(2.*CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(1.*XAD(I)))
     *          *TS(IP,K)
      GO TO 5000
C
   8  AA(K)= -2.*CZ(I,KM,M,L)/DZ(K)
      DD(K)= (RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L)
     *         )/DT+2.*(CZ(I,KM,M,L)+CZ(I,K,3,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,3,L)/DZ(K)
      BB(K)= ((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*
     *     SH(I,K,3,L))/DT-2.*(CX(I,K,M,L)+CX(I,K,3,L))/DX(I)
     *     -CYL*(CX(I,K,M,L)+CX(I,K,3,L))/(1.*XAD(I)))*TS(I,K)
     *      +(2.*(CX(I,K,M,L)+CX(I,K,3,L))/DX(I)+CYL*(CX(I,M,M,L)+
     *          CX(I,K,3,L))/(1.*XAD(I)))*TS(IP,K)
      GO TO 5000
C
   9  AA(K)= 0.0
      DD(K)= RO(I,K,M,L)*SH(I,K,M,L)/DT+2.*(CZ(I,K,M,L)
     *         +HT(I))/DZ(K)
      CC(K)= -2.*CZ(I,K,M,L)/DZ(K)
      BB(K)= (RO(I,K,M,L)*SH(I,K,M,L)/DT-2.*CX(I,K,M,L)/DX(I)
     *          -CYL*CX(I,K,M,L)/(1.*XAD(I)))*TS(I,K)
     *         +(2.*CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(1.*XAD(I)))*
     *         TS(IP,K)+2.*HT(I)/DZ(K)*TAMB
      GO TO 5000
C
  10  AA(K)= -2.*CZ(I,KM,3,L)/CZ(K)
      DD(K)= RO(I,K,3,L)*SH(I,K,3,L)/DT+2.*(CZ(I,KM,3,L)
     *         +HB(I,L))/DZ(K)
      CC(K)= 0.0
      BB(K)= (RO(I,K,3,L)*SH(I,K,3,L)/DT-2.*CX(I,K,3,L)/DX(I)
     *          -CYL*CX(I,K,3,L)/(1.*XAD(I)))*TS(I,K)
     *         +(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/
     *         (1.*XAD(I)))*TS(IP,K)
     *          +2.*HB(I,L)/DZ(K)*TAMB
      GO TO 5000
C
  11  IF(TIME.GE.TGAP) GO TO 911
      AA(K)= -(CZ(I,KM,M,L)+CZ(I,KM,3,L))/DZ(K)
      DD(K)= (RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)
     *         *SH(I,K,3,L))/DT+(CZ(I,KM,M,L)+CZ(I,KM,3,L)
     *         +CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K)
      CC(K)= -(CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K)
      BB(K)=(2.*CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/XAD(I))
     *      *TS(IM,K)+((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*
     *     SH(I,K,3,L))/DT-2.*(CX(IM,K,M,L)+CX(I,K,3,L))/DX(I)
     *      -CYL*(CX(I,K,3,L)-CX(IM,K,M,L))/XAD(I))*TS(I,K)
     *  +(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))*TS(IP,K)
      GO TO 5000
 911  AA(K)= -CZ(I,KM,3,L)/DZ(K)
      DD(K)=RO(I,K,3,L)*SH(I,K,3,L)/DT+(CZ(I,KM,3,L)
     *       +CZ(I,K,3,L))/DZ(K)
      CC(K)= -CZ(I,K,3,L)/DZ(K)
      BB(K)= 2.0*HG(M,L)/DX(I)*TS(IM,K)+
```

```
IV 3 LEVEL  21                    LINEZ                 DATE = 83179          13/34/53
     *    (RC(I,K,3,L)*SH(I,K,3,L)/DT-2.*(HG(K,L)+CX(I,K,3,L))
     *      /DX(I)-CYL*CX(I,K,3,L)/XAD(I))*TS(I,K)
     * +(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))*TS(IP,K)
      GO TO 5000
C
  12  AA(K)= -2.*CZ(I,KM,M,L)/DZ(K)
      DD(K)= (RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L)
     *      )/DT+2.*(CZ(I,KM,M,L)+CZ(I,K,3,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,3,L)/DZ(K)
      BB(K)= ((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL*
     *      (CX(IM,K,M,L)+CX(IM,K,3,L))/(2.*XAD(I)))*TS(IM,K)
     * +((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))/DT
     *  -(CX(IM,K,M,L)+CX(IM,K,3,L)+CX(I,K,M,L)+CX(I,K,3,L))
     *  /DX(I)-CYL*(CX(I,K,M,L)+CA(I,K,3,L)-CX(IM,K,M,L)-
     *   CX(IM,K,3,L))/(2.*XAD(I)))*TS(I,K)
     *     +((CX(I,K,M,L)+CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)+
     *       CX(I,K,3,L))/(2.*XAD(I)))*TS(IP,K)
      GO TO 5000
C
  13  IF(TIME.GE.TGAP) GO TO 513
      AA(K)= -(CZ(I,KM,M,L)+CZ(I,KM,3,L))/DZ(K)
      DD(K)= (RO(I,K,M,L)*SH(I,K,M,L)+3.*RO(I,K,3,L)*
     *       SH(I,K,3,L))/(2.*DT)+(CZ(I,KM,M,L)+CZ(I,KM,3,L)
     *      +2.*CZ(I,K,3,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,3,L)/DZ(K)
      BB(K)= ((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL*
     *      (CX(IP,K,M,L)+CX(IM,K,3,L))/(2.*XAD(I)))*TS(IM,K)
     * +((RO(I,K,M,L)*SH(I,K,M,L)+3.*RO(I,K,3,L)*SH(I,K,3,L)
     *  )/(2.*DT)-(CX(IM,K,M,L)+CX(IM,K,3,L)+2.*CX(I,K,3,L))/
     *    CX(I)-CYL*(2.*CX(I,K,3,L)-CX(IP,K,3,L)-CX(IM,K,3,L))/
     *    (2.*XAD(I)))*TS(I,K)
     *  +(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))*TS(IP,K)
      GO TO 5000
 513  AA(K)= -CZ(I,KM,3,L)/DZ(K)
      DD(K)=(3.*RO(I,K,3,L)*SH(I,K,3,L))/(2.*DT)+(CZ(I,KM,3,L)+
     *      2.*CZ(I,K,3,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,3,L)/DZ(K)
      BB(K)=((CX(IP,K,3,L)+HG(K,L))/DX(I)-CYL*CX(IM,K,3,L)/
     *     (2.*XAD(I)))*TS(IM,K)+((3.*RO(I,K,3,L)*SH(I,K,3,L))/
     *    (2.*DT)-(CX(IM,K,3,L)+HG(K,L)+2.*CX(I,K,3,L))/DX(I)
     *    -CYL*(2.*CX(I,K,3,L)-CX(IM,K,3,L))/(2.*XAD(I)))*TS(I,K)
     *  +(2.*CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/XAD(I))*TS(IP,K)
      GO TO 5000
C
  14  IF(TIME.GE.TGAP) GO TO 514
      AA(K)= 0.0
      DD(K)=(RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))
     *     /(2.*DT)+(HTC+HTM+CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K)
      CC(K)= -(CZ(I,K,M,L)+CZ(I,K,3,L))/DZ(K)
      BB(K)=(CX(IP,K,M,L)/DX(I)-CYL*CX(IP,K,M,L)/(2.*XAD(I)))*
     *     TS(IM,K)+((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*
     *   SH(I,K,3,L))/(2.*DT)-(CX(IP,K,M,L)+CX(I,K,3,L))/DX(I)
     *     -CYL*(CX(I,K,3,L)-CX(IM,K,M,L))/(2.*XAD(I)))*TS(I,K)
     * +(CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(2.*XAD(I)))*TS(IP,K)
     *     +(HTC+HTM)/DZ(K)+TAMB
      GO TO 5000
 514  AA(K)= 0.0
      DD(K)=RO(I,K,3,L)*SH(I,K,3,L)/(2.*DT)+(HT(I)
     *     +CZ(I,K,3,L))/DZ(K)
      CC(K)= -CZ(I,K,3,L)/DZ(K)
      BB(K)= HG(K,L)/DX(I)*TS(IM,K)+
     * (RO(I,K,3,L)*SH(I,K,3,L)/(2.*DT)-(HG(K,L)+CX(I,K,3,L))
```

```
(V 3 LEVEL  21                    LINKZ                 DATE = 83178          14/34/53
     *       /DX(I)-CYL*CX(I,K,3,L)/(2.*XAC(I)))*TS(I,K)
     *   +(CX(I,K,3,L)/DX(I)+CYL*CX(I,K,3,L)/(2.*XAD(I)))*TS(IP,K)
     *         +HT(I)/DZ(K)*TAMB
      GO TO 5006
C
   15 IF(TIME.GE.TGAP) GO TO 515
      AA(K)= -CZ(I,KM,M,L)/DZ(K)
      DD(K)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CZ(I,KM,M,L)
     *       +CZ(I,K,M,L))/DZ(K)
      CC(K)= -CZ(I,K,M,L)/DZ(K)
      BB(K)= (CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAC(I)))
     *       *TS(IM,K)+(RO(I,K,M,L)*SH(I,K,M,L)/DT
     *       -(CX(IM,K,M,L)+CX(I,K,M,L))/DX(I)
     *       -CYL*(CX(I,K,M,L)-CX(IM,K,M,L))/(2.*XAD(I)))*TS(I,K)
     * +(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAD(I)))*TS(IP,K)
      GO TO 5006
  515 AA(K)= -CZ(I,KM,M,L)/DZ(K)
      DD(K)= RO(I,K,M,L)*SH(I,K,M,L)/DT+(CZ(I,KM,M,L)
     *       +CZ(I,K,M,L))/DZ(K)
      CC(K)= -CZ(I,K,M,L)/DZ(K)
      BB(K)= (2.*CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/XAC(I))*
     *  TS(IM,K)+(RO(I,K,M,L)*SH(I,K,M,L)/DT-2.*(CX(IM,K,M,L)
     *      +HG(M,L))/DX(I)+CYL*CX(IM,K,M,L)/XAD(I))*TS(I,K)
     *       +2.*HG(M,L)/DX(I)*TS(IP,K)
      GO TO 5006
C
   16 IF(TIME.GE.TGAP) GO TO 516
      AA(K)= -2.0*CZ(I,KM,M,L)/DZ(K)
      DD(K)= (RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L)
     *       )/DT+2.*(CZ(I,KM,M,L)+CZ(I,K,3,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,3,L)/DZ(K)
      BB(K)= ((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL*
     *  (CX(IM,K,M,L)+CX(IM,K,3,L))/(2.*XAD(I)))*TS(IM,K)
     *    +((RO(I,K,M,L)*SH(I,K,M,L)+RO(I,K,3,L)*SH(I,K,3,L))
     *      /DT-(CX(IM,K,M,L)+CX(IM,K,3,L)+CX(I,K,M,L)+
     *      CX(I,K,3,L))/DX(I)-CYL*(CX(I,K,M,L)+CX(I,K,3,L)
     *      -CX(IM,K,M,L)-CX(IM,K,3,L))/(2.*XAD(I)))*TS(I,K)
     *      +((CX(I,K,M,L)+CX(I,K,3,L))/DX(I)+CYL*(CX(I,K,M,L)+
     *        CX(I,K,3,L))/(2.*XAD(I)))*TS(IP,K)
      GO TO 5006
  516 AA(K)= -CZ(I,KM,M,L)/DZ(K)
      DD(K)= (RO(I,K,M,L)*SH(I,K,M,L)+2.*RO(I,K,3,L)*
     * SH(I,K,3,L))/(2.*DT)+(CZ(I,KM,M,L)+2.*CZ(I,K,3,L))/CZ(K)
      CC(K)= -(2.*CZ(I,K,3,L))/DZ(K)
      BB(K)=((CX(IM,K,M,L)+CX(IM,K,3,L))/DX(I)-CYL*(CX(IM,K,M,L)
     *       +CX(IM,K,3,L))/(2.*XAD(I)))*TS(IM,K)
     * +((RO(I,K,M,L)*SH(I,K,M,L)+2.*RO(I,K,3,L)*SH(I,K,3,L))
     *   /(2.*DT)-(CX(IM,K,M,L)+CX(IM,K,3,L)+CX(I,K,3,L)+
     *   HG(K,L))/DX(I)-CYL*(CX(I,K,3,L)-CX(IP,K,M,L)-
     *  CX(IP,K,3,L))/(2.*XAD(I)))*TS(I,K)
     *       +((CX(I,K,3,L)+HG(M,L))/DX(I)+CYL*CX(I,K,3,L)/
     *       (2.*XAD(I)))*TS(IP,K)
      GO TO 5006
C
   17 IF(TIME.GE.TGAP) GO TO 517
      AA(K)= 0.0
      DD(K)= RO(I,K,M,L)*SH(I,K,M,L)/DT+2.*(HT(I)
     *       +CZ(I,K,M,L))/DZ(K)
      CC(K)= -2.*CZ(I,K,M,L)/DZ(K)
      BB(K)= (CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
     *    *TS(IM,K)+(RO(I,K,M,L)*SH(I,K,M,L)/DT-(CX(IM,K,M,L)
     *       +CX(I,K,M,L))/DX(I)
```

```
V  0 LEVEL  21                    LINEZ                 DATE = 83179          19/34/53
     *              -CYL*(CX(I,K,M,L)-CX(IM,K,M,L))/(2.*XAD(I)))*TS(I,K)
     * +(CX(I,K,M,L)/DX(I)+CYL*CX(I,K,M,L)/(2.*XAC(I)))*TS(IP,K)
     *              +2.*HT(I)/DZ(M)*TAMB
      GO TO 55.C
  317 AA(K)= 0.C
      DD(K)=RO(I,K,M,L)*SH(I,K,M,L)/(2.*DT)+(HT(I)
     *       +CZ(I,K,M,L))/DZ(K)
      CC(K)= -CZ(I,K,M,L)/DZ(K)
      BB(K)= (CX(IM,K,M,L)/DX(I)-CYL*CX(IM,K,M,L)/(2.*XAD(I)))
     *       *TS(IM,K)+(RO(I,K,M,L)*SH(I,K,M,L)/(2.*DT)
     *       -(CX(IP,K,M,L)+HG(K,L))/
     *          DX(I)+CYL*CX(IM,K,M,L)/(2.*XAC(I)))*TS(I,K)
     *         +HG(K,L)/DX(I)*TS(IP,K)
     *          +HT(I)/DZ(K)*TAMB
C
 5..0 CONTINUE
      CALL TRID(AA,DD,CC,BB,NBAR)
      DO 30. K=1,NBAR
  30. T(I,K)=RR(K)
 55C. CONTINUE
      RETURN
      END
```

```
IV 3 LEVEL  21                MAIN                DATE = 83179          18/34/53
C
C
      SUBROUTINE TRIC(A,D,C,B,N)
C
C     ------------------------------------------------------------------
C     |                                                                |
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
C     |                                                                |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
C     |                                                                |
C     |  UNIVERSITY OF OTTAWA.                                         |
C     |                                                                |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
C     |                                                                |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
C     |                                                                |
C     |  DR. M. SALCUDEAN.                                             |
C     |                                                                |
C     ------------------------------------------------------------------
C
C     COMPUTE THE SOLUTIONS FOR THE SYSTEM OF LINEAR EQUATION
C     BY THE "TDMA".
      COMMON /GEOM/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TFOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,ROLIQ,
     * ROSOL,ROMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAS,KBAS,IFACE,KFACE,IFR,IFP,KFR,KFP,IC(25,50),
     * ICODE,ICYCLE,CT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DIMENSION A(50),D(50),C(50),B(50)
C
      IF(N.GT.1) GO TO 10
      B(1)=B(1)/D(1)
      RETURN
C
   10 DO 110 I=2,N
      RATIO= -A(I)/D(I-1)
      D(I)= D(I)+RATIO*C(I-1)
  110 B(I)= B(I)+RATIO*B(I-1)
      B(N)= B(N)/D(N)
      DO 120 NP=2,N
      I=N-NP+1
  120 B(I)= (B(I)-C(I)*B(I+1))/D(I)
      RETURN
      END
```

```
FTN 3 LEVEL  21                    MAIN                    DATE = 83179          19/34/53
C
C
      FUNCTION RO(I,K,P,L)
C
C     ------------------------------------------------------------------
C     |                                                                |
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
C     |                                                                |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
C     |                                                                |
C     |  UNIVERSITY OF OTTAWA.                                         |
C     |                                                                |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
C     |                                                                |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
C     |                                                                |
C     |  DR. M. SALCUDEAN.                                             |
C     |                                                                |
C     ------------------------------------------------------------------
C
C     SUPPLY THE REQUIRED DENSITY.
C
      COMMON /GEOM/ DX(25),XAC(25),DZ(50),ZAC(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TFOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,ROLIQ,
     * ROSOL,ROMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFP,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      GO TO (100,200,300,400),M
  100 RO=ROSOL
      RETURN
  200 RO=ROLIQ
      RETURN
  300 RO=ROMOLD
      RETURN
  400 RO=ROLIQ
      RETURN
      END
```

```
IV 3 LEVEL  21                    MAIN                    DATE = 83179          13/34/52
C
C
      FUNCTION SH(I,M,N,L)
C
C     ****************************************************************
C     *                                                              *
C     *  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:            *
C     *                                                              *
C     *  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,     *
C     *                                                              *
C     *  UNIVERSITY OF OTTAWA.                                       *
C     *                                                              *
C     *  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO    *
C     *                                                              *
C     *  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:             *
C     *                                                              *
C     *  DR. M. SALCUDEAN.                                           *
C     *                                                              *
C     ****************************************************************
C
C     SPECIFIC HEAT CALCULATED WITH TEMPERATURE (IN  F )
C     GIVES  SH  IN  BTU/LB.F   AND THEN CONVERT TO
C     SI UNITS  IN   J/KG.C
C
      COMMON /GEOM/ DX(25),XAC(25),DZ(50),ZAC(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TFOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RCLIQ,
     * RCSOL,RCMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      GO TO (1,2,3),L
    1 TE1= T(I,M)*1.8+32.0
      GO TO 99
    2 TE1= TN(I,M)*1.8+32.0
      GO TO 99
    3 TE1= TS(I,M)*1.8+32.0
   99 CONTINUE
      GO TO (1[illegible],2[illegible],300,4[illegible]),M
  1[illegible] IF(TE1.GE.800.0) GO TO 101
      SH= 0.118+0.[illegible]*(TE1-2[illegible])
      GO TO 105
  101 IF(TE1.GE.1300.0) GO TO 102
      SH= 0.148+0.[illegible]*(TE1-8[illegible])
      GO TO 105
  102 IF(TE1.GE.18[illegible]) GO TO 103
      SH= 0.222-0.[illegible]*(TE1-13[illegible])
      GO TO 105
  103 SH= 0.268+0.[illegible]*(TE1-18[illegible])
  105 SH= SH*4186.8
      RETURN
C
  2[illegible] SH= 0.21
      SH= SH*4186.8
      RETURN
C
  3[illegible] IF(TIME.GT.0.3) GO TO 333
```

```
FN G LEVEL  21                    SH                    DATE = 83179          18/34/53

      TE1=TPOLD*1.8+32.0
  333 IF(TE1.GE.800.0) GO TO 3.1
      SH= 0.110+0.000059*(TE1-200.0)
      GO TO 3.5
  3.1 IF(TE1.GE.1300.0) GO TO 3.2
      SH= 0.148+0.000148*(TE1-800.0)
      GO TO 3.5
  3.2 IF(TE1.GE.1600.0) GO TO 3.3
      SH= 0.222-0.000108*(TE1-1300.0)
      GO TO 3.5
  3.3 SH= 0.168+0.000381*(TE1-1800.0)
  3.5 SH= SH*4186.8
      RETURN
C
  4.0 IF(TE1.GE.2600.0) GO TO 4.1
      SH= 0.234+0.00179*(TE1-2400.0)
      GO TO 4.5
  4.1 SH= 2.835-0.00576*(TE1-2639.0)
  4.5 SH= SH*4186.8
      RETURN
      END
```

```
IV G LEVEL  21                    MAIN                    DATE = 83179          10/34/53
C
C
      FUNCTION COND(I,K,M,L)
C     ------------------------------------------------------------------
C     |                                                                |
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:              |
C     |                                                                |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,       |
C     |                                                                |
C     |  UNIVERSITY OF OTTAWA.                                         |
C     |                                                                |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO      |
C     |                                                                |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:               |
C     |                                                                |
C     |  DR. M. SALCUDEAN.                                             |
C     ------------------------------------------------------------------
C
C     CONDUCTIVITY CALCULATED USING TEMPERATURE ( IN  F )
C     GIVES  COND  IN  BTU/FT.HR.F   AND CONVERT TO
C     SI UNITS  IN  W/M**2.C
C
      COMMON /GEOM/ DX(25),XAC(25),DZ(5.),ZAC(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,5)),TN(25,50),TROOM,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STEAN,ROLIQ,
     * ROSOL,RCMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICOUP,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
C
      GO TO (1,2,3),L
    1 TE1= T(I,K)*1.8+32.0
      GO TO 99
    2 TE1= TN(I,K)*1.8+32.0
      GO TO 99
    3 TE1= TS(I,K)*1.8+32.0
   99 CONTINUE
      GO TO (1[0,2[0,3[0,4[0),M
  1[0 IF(TE1.GE.1640.0) GO TO 101
      COND= 28.58-0.0[0742*(TE1-1[0.0)
      GO TO 105
  101 COND= 17.3+0.0[0[4*(TE1-2640.0)
  105 COND= COND*1.7307
      RETURN
C
  2[0 COND= 15.0
      COND= COND*1.7307
      RETURN
C
  3[0 IF(TIME.GT.0.0) GO TO 333
      TE1=TMOLD*1.8+32.0
  333 IF(TE1.GE.1640.0) GO TO 301
      COND= 28.58-0.0[0742*(TE1-1[0.0)
      GO TO 305
  301 COND= 17.3+0.0[0124*(TE1-1640.0)
```

```
LEVEL  21                    COND                DATE = 83179          18/34/53

  303   CONDR CONDC*1.7307
        RETURN
C
  4..   CONDR 19.5-..0369*(TE1-2640.0)
        CONDR CONDC*1.7307
        RETURN
        END
```

```
IV G LEVEL  21                MAIN              DATE = 83179          18/34/53
C
      SUBROUTINE PRINT(NOS)
C     ------------------------------------------------------------------
C     |  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:               |
C     |  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,        |
C     |  UNIVERSITY OF OTTAWA.                                          |
C     |  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO       |
C     |  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:                |
C     |  DR. M. SALCUDEAN.                                              |
C     ------------------------------------------------------------------
C
C     PRINT OUT THE TEMPERATURE FIELD AND ANY REQUIRED PHYSICAL
C     CHARACTERS.
C
      COMMON /GEOM/ DX(25),XAC(25),DZ(50),ZAC(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPOUR,TMELT,TAMB,TLIQ,TSOL,
     * TPOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,RGLIQ,
     * RGSOL,RGMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFM,KFP,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      DIMENSION A2(25,50),S1(25,50),C1(25,50),M1(25,50)
      DIMENSION HS1(50),HB1(25),HG1(50)
C
C
      IF(NOS.NE.3) GO TO 4
      WRITE(6,121) ICYCLE
  121 FORMAT('1',10X,'( AFTER SUBROUTINE LINEX )',2X,'CYCLE=',I5,2X,
     * //)
      WRITE(6,199) (I,I=1,IBAR)
      DO 77 K=1,KBAR
   77 WRITE(6,201) K,(TS(I,K),I=1,IBAR)
      RETURN
    4 IF(NOS.NE.4) GO TO 55
      WRITE(6,122) ICYCLE
  122 FORMAT('1',10X,'( AFTER SUBROUTINE LINEZ )',2X,'CYCLE=',I5,2X,
     *  //)
      WRITE(6,199) (I,I=1,IBAR)
      DO 88 K=1,KBAR
   88 WRITE(6,201) K,(T(I,K),I=1,IBAR)
      RETURN
   55 CONTINUE
C
      DO 111 K=1,KBAR
      DO 111 I=1,IBAR
      IF(TN(I,K).LE.TSOL) M=1
      IF(TN(I,K).GE.TLIQ) M=2
      IF(TN(I,K).GT.TSOL.AND.TN(I,K).LT.TLIQ) M=4
```

```
V 3 LEVEL  21                    PRINT                DATE = 83179          10/34/53

      IF(I.GT.IFACE) M=3
      IF(K.GT.KFACE) M=3
      MA(I,K)=M
      HS1(K)= HS(K,2)
      HB1(I)= HB(I,2)
      HG1(K)= HG(K,2)
      R1(I,K)=RC(I,K,M,2)
      S1(I,K)=SH(I,K,M,2)
  111 C1(I,K)=COND(I,K,M,2)
      WRITE(6,190) ICYCLE,TIME
  190 FORMAT('1',20X,'TEMPERATURE DISTRIBUTION FOR CYCLE NO.=',I7,
     * 3X,'  TIME= ',F12.5,'  SEC3.'//)
      WRITE(6,199) (I,I=1,IBAR)
  199 FORMAT(//,7X,I2,19(4X,I2))
      DO 200 K=1,KBAR
      WRITE(6,201) K,(TN(I,K),I=1,IBAR)
      WRITE(6,202) (R1(I,K),I=1,IBAR)
C     WRITE(6,203) (IC(I,K),I=1,IBAR)
C     WRITE(6,204) (R1(I,K),I=1,IBAR)
C     WRITE(6,205) (S1(I,K),I=1,IBAR)
      WRITE(6,206) (C1(I,K),I=1,IBAR)
  201 FORMAT(///,' T',I2,1X,20(F6.0))
  202 FORMAT(' R ',4X,I2,19(4X,I2))
  203 FORMAT(' IC',4X,I2,19(4X,I2))
  204 FORMAT(' RC',2X,20(F6.2))
  205 FORMAT(' SH',2X,20(F6.1))
  206 FORMAT(' CO',2X,20(F6.1))
  200 CONTINUE
      DO 22. K=1,KBAR
  220 WRITE(6,221) K,HS1(K)
      DO 230 I=1,IBAR
  230 WRITE(6,231) I,HB1(I),I,HT(I)
  221 FORMAT(5X,'HS(',I2,')=',F8.3)
  231 FORMAT(5X,'HB(',I2,')=',F8.3,5X,'HT(',I2,')=',
     * F8.3)
      IF(TIME.LT.TGAP) RETURN
      DO 240 K=1,KFACE
  240 WRITE(6,241) K,HG1(K)
  241 FORMAT(/,5X,'HG(',I2,')=',F8.3)
C
      RETURN
      END
```

```
IV G LEVEL  21                MAIN              DATE = 83179          13/34/53
C
C
      SUBROUTINE TAPLOT
C*****************************************************************
C*                                                               *
C*  COPYRIGHT OF THIS SUBPROGRAM IS THE PROPERTY OF:             *
C*                                                               *
C*  DR. M. SALCUDEAN, DEPARTMENT OF MECHANICAL ENGINEERING,      *
C*                                                               *
C*  UNIVERSITY OF OTTAWA.                                        *
C*                                                               *
C*  INFORMATION CONTAINED HEREIN SHALL NOT BE TRANSMITTED TO     *
C*                                                               *
C*  ANY PARTIES WITHOUT EXPRESSED AUTHORIZATION OF:              *
C*                                                               *
C*  DR. M. SALCUDEAN.                                            *
C*                                                               *
C*****************************************************************
C
      COMMON /GRID/ DX(25),XAD(25),DZ(50),ZAD(50),FX(25,50),
     * FZ(25,50)
C
      COMMON /TEMP/ T(25,50),TN(25,50),TPOUR,TMELT,TAMB,TLIQ,TSOL,
     * TMOLD,TS(25,50)
C
      COMMON /PROP/ EMIS,STFAN,ROLIQ,
     * ROSOL,ROMOLD,HFUS,HT(25),
     * HTC,HTM
C
      COMMON /VAR/ IBAR,KBAR,IFACE,KFACE,IFM,IFP,KFP,KFF,IC(25,50),
     * ICODE,ICYCLE,DT,TSTOP,TPRINT,TPLOT,TIME,TGAP,CYL
C
      SCX=18.0/XAD(IBAR)
      SCZ=SCX
      CALL SYMBOL(1.0,9.0,.35,'TEMPERATURE DISTRIBUTION, TIME= ',0.,32)
      TT=TIME
      CALL NUMBER(999.,999.,.35,TT,90.,2)
      CALL SYMBOL(999.,999.,.35,' SECS.',90.,6)
C
      CALL PLOT(3.0,5.0,-3)
C
      CALL PLOT(-1.0,0.0,3)
      CALL PLOT(ZAD(20)*SCZ,0.0,2)
      CALL PLOT(ZAD(24)*SCZ,0.0,3)
      CALL PLOT(ZAD(32)*SCZ,0.0,2)
      CALL PLOT(ZAD(36)*SCZ,0.0,3)
      CALL PLOT(ZAD(KBAR)*SCZ+1.0,0.0,2)
      CALL PLOT(ZAD(KBAR)*SCZ,0.0,3)
C
      CALL PLOT(ZAD(KBAR)*SCZ,XAD(IBAR)*SCX,2)
      CALL PLOT(0.0,XAD(IBAR)*SCX,2)
      CALL PLOT(0.0,0.0,2)
C
      CALL PLOT(ZAD(KFACE)*SCZ,0.0,3)
      CALL PLOT(ZAD(KFACE)*SCZ,XAD(IFACE)*SCX,2)
      CALL PLOT(0.0,XAD(IFACE)*SCX,2)
C
      DO 300 K=1,KBAR
      DO 300 I=1,IBAR
      TEMP=TN(I,K)
      CALL SYMBOL(ZAD(K)*SCZ,XAD(I)*SCX,.2,5,90.,-1)
      CALL NUMBER(ZAD(K)*SCZ-.1,XAD(I)*SCX-.4,.2,TEMP,90.,-1)
```

```
LEVEL  21                    TNPLOT              DATE = 83178          18/34/53
600   CONTINUE
      CALL PLOTIZAD(KSAR)*SCZ*3.,-5.,-3)
      RETURN
      END
C
//GO.SYSIN DD *
 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.008 0.008
 0.020 0.020 0.020 0.020 0.020 0.020 0.020 0.020
 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023
 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023
 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023
 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023
/*
```

Δz Δx